# Tendencias en los índices de temperatura y precipitación extrema para Galicia en el periodo 1960-2017

Trabajo Fin de Grado Grado de Ciencias Ambientales

Marta Rilo Martínez

**Dra.** Mª de las Nieves Lorenzo González, profesora titular del área de Física de la Tierra, del departamento de Física Aplicada, de la Universidad de Vigo

INFORMA:

Que el trabajo titulado " Tendencias en los índices de temperatura y precipitación extrema para Galicia en el periodo 1960-2017" presentado por Da. Marta Rilo Martínez, con D.N.I.: 49203497B, ha sido realizado bajo mi dirección en el departamento de Física Aplicada de la Universidad de Vigo, y autorizo su presentación como Trabajo Fin de Grado del Grado de Ciencias Ambientales al considerarlo apto para ser defendido.

Ourense, 19 de Junio de 2019

Da. Ma de las Nieves Lorenzo González

## **RESUMEN**

A lo largo de las últimas décadas la preocupación por el cambio climático y sus posibles impactos sobre la población y los ecosistemas ha ido en aumento. Uno de sus efectos más relevantes son los cambios en la frecuencia e intensidad de los fenómenos extremos. La comprensión de las variaciones que se están produciendo en relación a este tipo de eventos supone una gran mejora de la capacidad humana de adaptarse al problema y mitigar sus efectos. Por este motivo, los estudios más recientes se centran en el desarrollo de índices, que permiten analizar los distintos fenómenos extremos. Éstos índices se han aplicado a diferentes escalas, temporales y espaciales observándose que los resultados varían mucho de unas escalas a otras. Es por ello que se hace necesario el cálculo de los índices extremos a nivel regional para mejorar la capacidad de resiliencia y adaptación de las diferentes poblaciones locales.

El objetivo de este trabajo es el análisis de las tendencias en los índices extremos de temperatura y precipitación en Galicia a partir de datos recogidos por estaciones meteorológicas de Meteogalicia y AEMET durante el periodo 1960-2017. Para ello se han escogido las estaciones meteorológicas que tienen las series de datos de precipitación y temperatura más homogéneas y de mejor calidad, y se han seleccionado los índices extremos recomendados por el ETCCDI (Expert Team on Climate Change Detection and Indices) más representativos en el área de estudio. Tras la validación de los datos y su homogeneización, se realizó el cálculo de los índices anual y estacionalmente analizando e interpretando los resultados obtenidos.

# ÍNDICE

1. In	trod	ucción	1
1.1.	Cli	ma y variabilidad climática	1
1.2.	Cai	mbio climático antropogénico	3
1.3.	ĺnd	ices extremos	5
1.4.	Áre	ea de estudio	7
1.5.	Ob	jetivos	. 10
2. Da	atos	y métodos	11
2.1.	Dat	tos	. 11
	.1.	Datos de precipitación y temperatura	
2.1	.2.	Índices extremos	
2.2.	Mé	todos	. 17
		Homogeneización de las series de temperatura y precipitación	
	2.2.	Tendencias	
3. Re	sul	tados	23
3.1.	ĺnd	ices extremos de temperatura	. 23
	.1.	Índice TX90	
3.1	.2.	Índice TX10	26
3.1	.3.	Índice TN90	28
3.1	.5.	Índice TXx	32
3.1	.6.	Índice TNn	34
3.1	.7.	Índice FD	36
3.1	.8.	Índice SU	38
3.1	.9.	Índice WSDI	40
3.1	.10.	Índice CSDI	42
3.2.	ĺnd	ices extremos de precipitación	. 44
3.2	2.1.	Índice PrecTot	44
3.2	2.2.	Índice SDII	46
3.2	2.3.	Índice R10 mm	48
3.2	2.4.	Índice R90p	50
3.2	2.5.	Índice R95pTot	52

	3.2.6.	Índice RX1D	. 54
	3.2.7.	Índice RX5D	. 56
	3.2.8.	Índice CDD	. 58
	3.2.9.	Índice CWD	. 60
4.	Discus	sión y conclusiones	63
4.	.1. Índ	ices de temperatura	63
4.	.2. Índ	ices de precipitación	65
RE	FEREN	ICIAS	69
ΑN	EXO A	· · · · · · · · · · · · · · · · · · ·	73
Α	.1 Cál	culo anual de los Índices de temperatura extrema	73
Α	.2 Cál	culo estacional de los Índices de temperatura extrema	87
Α	.3 Cál	culo anual de los Índices de precipitación extrema 1	108
Α	.4 Cál	culo estacional de los Índices de precipitación extrema 1	120

### 1. Introducción

#### 1.1. Clima y variabilidad climática

Generalmente, el clima se define como la descripción estadística en términos de la media y la variabilidad de un conjunto de parámetros relevantes durante un periodo de tiempo determinado que puede oscilar en escalas temporales de varios meses a miles o millones de años. Estos parámetros son habitualmente variables de superficie como temperatura, precipitación, humedad, viento... La duración mínima del periodo de tiempo que se recomienda para definir el clima de una región es, de acuerdo con la Organización Meteorológica Mundial (OMM), de 30 años (IPCC, 2013).

En general, el clima es el resultado de la existencia de múltiples y complejas interacciones que se establecen entre los cinco componentes principales que constituyen el sistema climático: atmósfera, hidrosfera, criosfera, litosfera y biosfera. El elevado grado de interrelación existente entre estos sistemas es la causa de que un cambio en cualquiera de ellos tenga implicaciones a escala global.

Los factores que condicionan y determinan el clima son numerosos y de muchos tipos, juntos dan lugar a la variabilidad climática, que puede ser definida como las variaciones del clima respecto al estado promedio y otros estadísticos como

la desviación estándar o la ocurrencia de extremos, a escalas temporales y espaciales mayores que las de los eventos meteorológicos individuales.

Cuando la variabilidad es debida a procesos naturales internos del sistema climático se denomina variabilidad interna, mientras que cuando es originada por acción de procesos externos, ya sean naturales o antropogénicos, se denomina variabilidad externa (IPCC, 2013).

La variabilidad interna es debida a los forzamientos que inducen cambios en el sistema climático a la vez que son integrantes del mismo, como por ejemplo el fenómeno de El Niño (El Niño South Oscillation), mientras que la variabilidad externa se debe a la influencia un agente externo, como las erupciones volcánicas o las variaciones en la insolación originadas por cambios en los parámetros orbitales (forzamiento orbital). Todos estos son forzamientos naturales. También existe lo que se conoce como forzamientos antropogénicos que son producidos como resultado de la actividad humana, este sería el caso de los cambios en la composición atmosférica inducidos por acción del hombre.

Las escalas temporales y espaciales en las que se pueden producir estos forzamientos son extremadamente amplias. A corto plazo se producen ciertos ciclos como los ciclos diarios de calentamiento-enfriamiento de la superficie terrestre y el de las brisas tierra-océano; Las variaciones a medio plazo son aquellas que ocurren en escalas de varios meses a algunos años, correspondientes con ciclos anuales y estacionales como el ciclo anual del dióxido de carbono en la atmósfera o las oscilaciones de la luminosidad solar en ciclos de 11 años; Por otra parte, la variabilidad a largo plazo implica una escala de tiempo geológica. En esta categoría se pueden incluir por ejemplo los ciclos de Milankovitch que describen variaciones de los parámetros orbitales: oblicuidad del eje, que oscila entre los 21.5° y 24.5° (periodo de 41.000 años), excentricidad de la órbita (periodo de 100.000 años) y precesión del eje (periodo de 25.790 años), los movimientos de las placas litosféricas, que condiciona la orografía y la distribución de continentes y océanos, o la propia evolución biológica de los seres vivos (Naranjo y Pérez-Muñuzuri, 2006).

Teniendo esto en cuenta, no cabe duda de que el cambio climático es algo que ha estado presente a lo largo de los 4600 millones de años de historia del planeta

Tierra y que durante este tiempo se han ido sucediendo climas muy diferentes unos de otros.

La diferencia entre los cambios climáticos ocurridos en el pasado y el que se está observando en la actualidad es que mientras los anteriores fueron ocasionados por fenómenos naturales, este está siendo provocado o acelerado por la actividad de los seres humanos. Es por ello que se habla de cambio climático antropogénico.

#### 1.2. Cambio climático antropogénico

Tal y como hemos dicho en la sección anterior todos los estudios climáticos apuntan a que el calentamiento y los cambios que se observan en el clima de nuestro planeta desde la revolución industrial son debidos a una alteración del balance de radiación en el planeta. Esta alteración es debida principalmente, al aumento incesante de la concentración de gases de efecto invernadero, especialmente dióxido de carbono (CO<sub>2</sub>), que se ha registrado en la atmósfera desde mediados del siglo XIX, con la invención de la máquina de vapor. Es en esta época en la que el ser humano comienza a utilizar los combustibles fósiles para obtención de energía y como combustible en los transportes de forma descontrolada (IPCC, 2013).

La explicación de la relación entre el incremento de la concentración de dióxido de carbono en el aire y el incremento de temperatura puede ser explicado por la capacidad que poseen las partículas que componen la atmósfera para interactuar con la radiación que nos llega del Sol. Algunos de los componentes atmosféricos absorben energía a algunas longitudes de onda de las que componen la radiación solar, como el ozono que absorbe en la región del ultravioleta, mientras que otros, como los aerosoles, la reflejan. De esta forma, cualquier variación en la composición de la atmósfera o de la concentración de los gases que la integran, se reflejará a través de un cambio en su temperatura. Los gases de efecto invernadero (vapor de agua, dióxido de carbono, metano, etc.) tienen la capacidad de absorber la radiación infrarroja térmica que emite la superficie terrestre y evitan que toda la energía recibida se pierda rápidamente

al ser reemitida en forma de calor de vuelta al espacio. Gracias a esto es posible la vida en el planeta Tierra tal y como la conocemos, ya que la temperatura media atmosférica es de 15°C, aproximadamente 33°C superior a la que tendría si estos gases no estuvieran presentes en la atmósfera. Sin embargo, aunque el efecto invernadero natural es algo positivo, un brusco incremento de su concentración en la atmósfera resulta perjudicial ya que aumenta el calentamiento del planeta y modifica las relaciones entre los diferentes componentes del sistema climático.

De acuerdo con los cálculos del IPCC, la variación de la temperatura atmosférica ha registrado un incremento de 0,85°C en el periodo 1880-2012. Además, el periodo 1983-2012, se muestra como el más cálido de los últimos 1400 años en el hemisferio norte mostrando un incremento decadal constante. Este incremento de la temperatura no es un hecho aislado, sino que desde la década de los años 50 se han observado numerosos cambios sin precedentes. Algunos de estos cambios son el calentamiento de la atmósfera y los océanos, el incremento del nivel del mar, la reducción de la cubierta de nieve y hielo, y el aumento de la concentración de gases de efecto invernadero (IPCC, 2013).

Muchos de los problemas que nos afectan hoy en día, como la escasez de alimentos, la falta de disponibilidad de agua potable o la expansión de enfermedades, podrían verse agravados como consecuencia de estos cambios observados. Además, surgen nuevas amenazas como la desaparición de áreas costeras como consecuencia de la dilatación térmica del agua o el aumento de la tasa de extinción de especies. Todos estos cambios tendrán consecuencias a nivel socioeconómico, condicionando la calidad de vida de las personas.

Cabe destacar que no todas las variaciones en los parámetros ocurren a la misma velocidad ni de forma uniforme, sino que las tendencias varían dependiendo de cada región. Por ejemplo, en relación con la precipitación encontramos que en el continente europeo las tendencias apuntan a un incremento con respecto a inicios del siglo XX, sin embargo, en general en el sur de Europa y más concretamente en la Península Ibérica, la tendencia se muestra negativa (European Environmental Agency, 2017, IPCC, 2013). Ocurre lo mismo con otros parámetros. Es por ello por lo que es necesario realizar estudios locales, pues la capacidad de adaptación de las diferentes regiones dependerá de cómo les afecte el cambio climático en particular.

#### 1.3. Índices extremos

Todos estos cambios de temperatura, precipitación, nivel del mar, etc. son importantes desde el punto de vista medioambiental, pues pueden afectar a los ecosistemas de forma negativa al modificar los hábitats naturales dificultando la supervivencia de las especies animales y vegetales presentes en los mismos. Además, estos cambios pueden afectar a la frecuencia e intensidad fenómenos o eventos extremos causantes de desastres naturales, como inundaciones, sequías o incendios forestales. Es por ello que en los últimos años el estudio del cambio climático se ha centrado en las variaciones que se producen en fenómenos meteorológicos extremos.

Un fenómeno meteorológico extremo se define como aquel que es muy poco frecuente en un lugar concreto y en una época del año determinada, siendo considerado como muy poco frecuente aquel fenómeno con una probabilidad de ocurrencia por encima o por debajo de los percentiles 90 y 10 de su función de densidad de probabilidad, calculada a partir de las observaciones realizadas en una región determinada. Con esta definición, el clima extremo es algo que depende de las características de cada territorio y por tanto se define de forma muy distinta en cada uno de ellos, pudiendo encontrar diferencias importantes de unos territorios a otros. Las precipitaciones extremas, los huracanes, los tornados, las olas de frío o calor y las sequías son algunos ejemplos de este tipo de fenómenos (IPCC, 2013).

Con el objetivo de estudiar estos eventos se ha creado una serie de índices denominados índices extremos. Estos índices se calculan a partir de datos diarios de temperatura o precipitación por encima o por debajo de umbrales establecidos de acuerdo a diferentes parámetros físicos específicos. Los umbrales fijados para cada índice cambian de acuerdo a las condiciones climáticas de cada área de estudio, ya que no tendría sentido aplicar los mismos criterios para el análisis de temperaturas en un clima polar que en uno subtropical, por ejemplo.

Ya que las condiciones cambian enormemente dependiendo de la zona del planeta estudiada, los índices se formulan pensados para su aplicación concreta y muy pocos tienen validez de aplicación a nivel global (Zhang et al., 2011). De

este hecho surge la necesidad de establecer un criterio único que permita realizar comparaciones entre diferentes regiones y poder evaluar la situación global. Con el objetivo de solucionar este problema y poder estudiar la evolución de los extremos climáticos a escala planetaria, el ETCCDMI (Expert Team on Climate Change Detection, Monitoring and Indices), un equipo internacional de expertos en cambio climático con experiencia en numerosos campos relacionados con el clima, ha seleccionado un total de 27 índices de extremos de precipitación y temperatura que sí son aplicables de forma generalizada (Peterson, 2005). Estos índices fueron los utilizados en este trabajo para el análisis del clima de Galicia. Los índices utilizados y sus descripciones se concretan en el capítulo de Datos y Métodos. La elección de los mismos se ha hecho siguiendo la línea de trabajos previos (Rodrigo y Trigo, 2007; Gómez-Gesteira et al., 2011; Ramos et al., 2011; Santo et al., 2014) lo que nos permitirá llevar a cabo una comparativa entre los resultados encontrados en otros estudios similares y en nuestro trabajo.

Los valores medios son especialmente útiles para la detección de cambios a largo plazo, en cambio los índices de extremos climáticos son eficaces también en la detección de cambios a corto plazo (diarios, trimestrales o anuales), siendo esta su principal ventaja. Mientras un valor medio puede mantenerse constante el clima puede estar cambiando, ya que, si los valores extremos máximos y mínimos se acentúan más o menos por igual, la media puede permanecer invariable. Por ejemplo, si en una determinada región se incrementan las temperaturas máximas y disminuyen las mínimas, la media puede permanecer invariable a pesar de que el clima se esté extremando.

Por otro lado, aunque estos índices son eficaces en la medida de la variabilidad diaria de los extremos climáticos y permiten el seguimiento de los posibles cambios en cuanto a frecuencia y/o intensidad, ha de tenerse en cuenta que su eficacia depende directamente de la calidad de las series de datos utilizadas. Si las series no son lo suficientemente largas, están incompletas o no son homogéneas, pueden obtenerse resultados erróneos. Para asegurar resultados fiables, en este estudio se han seleccionado las estaciones de medida con las series más antiguas y uniformes, además de haber establecido una serie de

criterios, especificados en el apartado de Datos y métodos, que aseguren una buena homogeneidad de las series (Zhang et al., 2011).

De acuerdo con los estudios realizados en los últimos años, a escala global se han producido muchos cambios en los fenómenos meteorológicos extremos tanto en cuanto a su frecuencia como a su persistencia e intensidad (Yan et al., 2002; Rodrigo y Trigo, 2007). Es muy probable que el número de días y noches fríos se haya reducido a la vez que los días y noches cálidos han aumentado. En Europa, Asia y Australia la frecuencia de las olas de calor también ha crecido. Por otra parte, en cuanto a los eventos de precipitación extrema también se han detectado cambios, aunque estos no son uniformes en todo el planeta. En determinadas áreas su frecuencia ha mermado, sin embargo, en la mayor parte del planeta son ahora más frecuentes, por lo que el cómputo final indica un incremento global de este tipo de eventos (IPCC, 2013).

#### 1.4. Área de estudio

Las tendencias en el cambio climático no son uniformes y los cambios que se puedan producir en el clima a pequeñas escalas espaciales no tienen por qué coincidir con los que se observan teniendo en cuenta el cómputo global, de ahí la importancia de llevar a cabo estudios regionales que permitan adecuar a cada lugar las políticas de adaptación al cambio. El área de estudio en la que se centra este trabajo es la correspondiente al territorio que constituye la Comunidad Autónoma de Galicia, situada en el extremo noroeste de la Península Ibérica, al suroeste del continente europeo, entre los 42° y 44° de latitud norte y 9° y 7° de longitud oeste.

En relación con el balance energético, Galicia se encuentra en una región de transición entre las zonas del planeta que reciben grandes cantidades de energía y las que sufren déficit. Es por tanto una región que se ve muy afectada por los mecanismos atmosféricos responsables de la redistribución de la energía en el planeta (Figura 1.1).

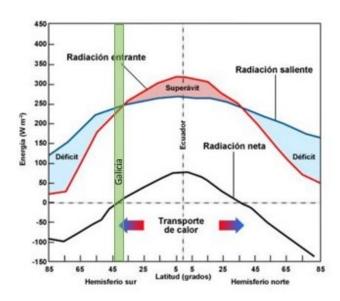


Figura 1.1. Balance radiativo en función de la latitud (Fuente: the COMET program).

Además, Galicia se encuentra bajo la influencia de los vientos del oeste de la célula de Ferrel que circulan a latitudes medias (entre 30 y 60° N, desde la zona de los trópicos hacia los polos) y constituye un punto de llegada de las borrascas atlánticas. Otro factor que influye en el clima de Galicia es el vórtice circumpolar, una corriente ondulatoria cerrada que se establece entorno al polo, en el límite entre la célula Polar y la de Ferrel, que actúa como mecanismo de transferencia de energía térmica, humedad y momento cinético entre las altas y las bajas latitudes (Figura 1.2.).

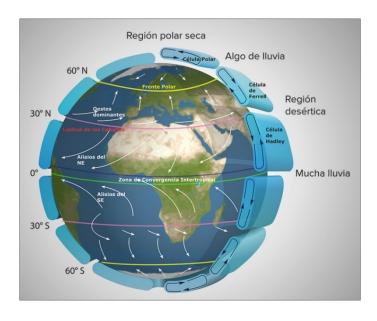


Figura 1.2. Sistema básico de la circulación general atmosférica. (Imagen modificada registrada bajo licencia de <u>CC BY-SA 3.0</u>).

En general, Galicia se ve afectada de forma indirecta por la corriente en chorro a través de la llegada de sistemas frontales asociados a ondas ciclónicas generadas a mayor latitud (perturbaciones ondulatorias del Frente Polar), aunque en ocasiones se producen situaciones en las que se ve afectada directamente por esta corriente. En estos casos se produce advección de aire oceánico que ocasiona lluvias generalizadas en todo el norte de la Península Ibérica.

Las masas de aire que llegan a Galicia poseen características térmicas e higrométricas muy distintas. Se distinguen tres tipos principales de masas de aire según su lugar de formación: polar, ártica y tropical. Las dos primeras son siempre frías y la última cálida. Además, estas masas serán secas o húmedas según se hayan originado sobre los continentes o el océano, respectivamente. Todo esto provoca que los tres tipos de masas de aire anteriores se subdividan en: polar marítimo o continental, ártico marítimo o continental y tropical marítimo o continental. En la Figura 1.3 se muestran las principales masas de aire que pueden afectar a nuestra región de estudio (Naranjo y Pérez-Muñuzuri, 2006).

Es posible identificar tres centros de acción en superficie responsables de la llegada de las distintas masas de aire que afectan al norte peninsular: La depresión de Islandia, el Anticiclón de las Azores y los Anticiclones térmicos centroeuropeos.

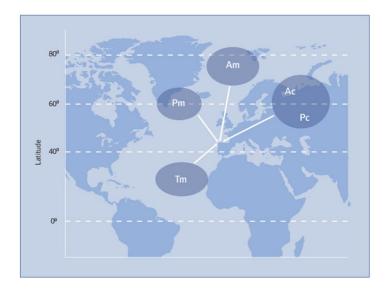


Figura 1.3. Tipos de masas de aire que afectan al noroeste de la península Ibérica: Ártica marítima (Am), Polar marítima (Pm), Tropical marítima (Tm), Ártica continental (Ac), Polar continental (Pc). Fuente: Naranjo y Pérez-Muñuzuri, 2006.

Por otra parte, otro factor a tener en cuenta es la morfología del terreno, que al ser compleja y variada origina diferencias climáticas dentro del territorio a meso y microescala (Martínez-Cortizas y Pérez-Alberti, 1999; Cruz et al., 2009).

#### 1.5. Objetivos

El objetivo de este trabajo es calcular y analizar una serie de índices extremos utilizando series de datos de precipitación y temperatura de la Comunidad Autónoma de Galicia que abarcan el periodo 1960-2014 para conocer los cambios en las condiciones climáticas extremas que se están produciendo en la región como consecuencia del cambio climático. Para ello se realizará:

- Una reconstrucción y validación de las series de datos de temperatura y precipitación procedentes de la red de estaciones meteorológicas de AEMET y de Meteogalicia para el periodo 1960-2014.
- Un estudio de las tendencias de los índices extremos de temperatura considerados a través de la aplicación del Test de Mann-Kendall.
- Un estudio de las tendencias de los índices extremos de precipitación considerados a través de la aplicación del Test de Mann-Kendall.

# 2. Datos y métodos

#### 2.1. **Datos**

#### 2.1.1. Datos de precipitación y temperatura

El cálculo de los índices de extremos climáticos en Galicia se realizó a partir de datos diarios de precipitación y temperatura medidos por un conjunto de 17 estaciones meteorológicas, distribuidas por toda el área de estudio. Aunque la mayoría de estas estaciones midieron datos de ambas variables, unas pocas tomaron exclusivamente datos de temperatura o precipitación. Considerándolas independientemente, en el caso de precipitación se utilizaron 13 estaciones y en el de temperatura 11. Dichas estaciones pertenecen a la Agencia Estatal de Meteorología (AEMET) y a Meteogalicia, que depende de la Xunta de Galicia.

Las series de datos utilizadas en este estudio fueron reconstruidas a partir de los datos procedentes de las dos redes de estaciones meteorológicas anteriormente citadas. Los datos de las primeras décadas pertenecen a las estaciones de AEMET y los de los últimos años a las de Meteogalicia. Tras una serie de análisis estadísticos que permitieron comprobar que es posible continuar las series de datos de las estaciones de AEMET con los datos de sus respectivas equivalentes de la red de Meteogalicia, se obtuvo para cada ubicación una serie de medidas

diarias que comprende el periodo 1960-2017 en el caso de las precipitaciones, y 1965-2014 en el de las temperaturas. Algunas de las estaciones abarcan un periodo más corto de tiempo, ya que entraron en funcionamiento más tarde. En la Tabla 2.1 se especifica la ubicación de las estaciones utilizadas para la medición de la temperatura junto con el periodo de tiempo de los datos registrados.

Tabla 2.1. Ubicación de las estaciones y longitud de las series de datos de temperatura.

Estación	Localización (WGS84)	Periodo
Allariz (AL)	-7.80, 42.18	1975 - 2014
As Pontes (AP)	-7.86,43.45	1965 - 2014
A Coruña (LC)	-8.42, 43.37	1965 - 2014
A Coruña Aeropuerto (AV)	-8.37, 43.31	1971 - 2014
Lavacolla (LV)	-8.41, 42.89	1965 - 2014
Lourizán (LO)	-8.65, 42.44	1965 - 2014
Lugo (LU)	-7.55, 43.00	1967 - 2014
Ourense (OU)	-7.86, 42.33	1973 - 2014
Paramos-Guillarei (PG)	-8.61 42.06	1969 - 2014
Vigo (VI)	-8.72, 42.23	1965 - 2014
Xinzo de Limia (XL)	-7.73, 42.08	1965 - 2014

La Tabla 2.2 indica la ubicación de las estaciones que tomaron datos de precipitación, así como el periodo de tiempo que abarca cada una.

Tabla 2.2. Ubicación de las estaciones y longitud de las series de datos de precipitación.

Estación	Localización (WGS84)	Periodo
Allariz (AL)	-7.80, 42.18	1960 - 2017
As Pontes (AP)	-7.86, 43.45	1960 - 2017
A Coruña (LC)	-8.42, 43.37	1960 - 2017
Ferrol (FE)	-8.19, 43.45	1960 - 2014
Fornelos (FO)	-8.45, 42.33	1977 - 2013
Lavacolla (LV)	-8.41, 42.89	1960 - 2017
Lugo (LU)	-7.55, 43.00	1966 - 2017
Lourizán (LO)	-8.65, 42.44	1960 - 2013
Ordes (O)	-8.43, 43.03	1960 - 2017
Ourense (OU)	-7.86, 42.33	1973 - 2017
Rozas (RO)	-7.46, 43.11	1986 - 2017
Sarria (SA)	-7.50, 42.85	1960 - 2017
Vigo (VI)	-8.72, 42.23	1960 - 2017

En el mapa de la Figura 2.1 podemos visualizar la localización de las 17 estaciones consideradas en el estudio.

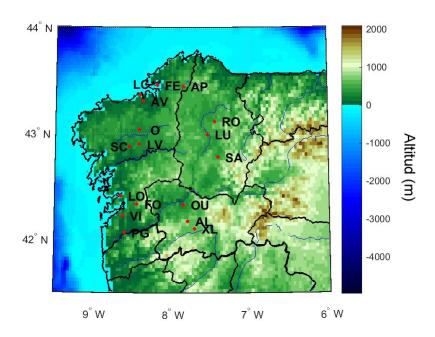


Figura 2.4. Área de estudio y ubicación de las 17 estaciones que midieron las variables de temperatura y precipitación utilizadas en el cálculo de índices extremos.

#### 2.1.2. Índices extremos

Los índices extremos tienen como objetivo la evaluación de los valores diarios de precipitación o temperatura registrados en relación a un umbral dado. Este umbral es relativo a cada región de estudio, ya que en el planeta existe una amplia variedad de climas y si los umbrales establecidos fuesen absolutos este tipo de índices carecería de validez en muchas regiones, y serían aptos solamente en las zonas para las que fueran diseñados (Zhang et al., 2011).

En este trabajo se han utilizado los índices extremos recomendados por Expert Team on Climate Change Detection and Indices (ETCCDI), un equipo que ofrece coordinación y colaboración internacional para la detección del cambio climático y el estudio de la variabilidad del clima, especializado en eventos extremos y patrocinado por la World Meteorological Organization (WMO) Commision for Climatology (CCI), el Climate Variability and Predictability Project (CLIVAR) y la Joint Commision for Oceanography and Marine Meteorology (JCOMM) (ETCCDI/CRD, 2013).

El conjunto de índices de temperatura extrema se recoge en la Tabla 2.3, los de precipitación extrema en la Tabla 2.4. En ambas tablas se ha incluido la definición de los índices, así como las unidades en las que se miden.

Tabla 2.3. Índices de temperatura extrema.

Índice	Definición	Unidades
TX90	Número de días con temperatura máxima superior al percentil 90.	días
TX10	Número de días con temperatura máxima inferior al percentil 10.	días
TN90	Número de días con temperatura mínima superior al percentil 90.	días
TN10	Número de días con temperatura mínima inferior al percentil 10.	días
TXx	Valor máximo de temperatura máxima diaria.	°C
TXn	Valor mínimo de temperatura máxima diaria.	°C
TNx	Valor máximo de temperatura mínima diaria.	°C
TNn	Valor mínimo de temperatura mínima diaria.	°C
FD	Días de helada. Número de días con temperatura mínima inferior a 0°C.	días
ID	Días de Hielo. Número de días con temperatura máxima inferior a 0°C.	días
SU	Días de calor. Número de días con temperatura máxima superior a 25°C.	días
TR	Noches tropicales. Número de noches con temperatura mínima superior a 20°C.	días
WDSI	Número de periodos cálidos, con al menos 6 días consecutivos de temperatura máxima superior al percentil 90.	Nº de periodos
CSDI	Número de periodos fríos, con al menos 6 días consecutivos de temperaturas mínimas inferiores al percentil 10.	Nº de periodos
GSL	Longitud de la estación de crecimiento. Número de días al año con temperatura media superior a 5°C.	días
DTR	Rango de temperaturas diurnas. Media de la diferencia entre las temperaturas máxima diaria y mínima diaria.	°C
ETR	Rango de temperaturas extremas. Diferencia entre el valor máximo de temperatura máxima (TXx) y el mínimo de temperatura mínima (TNn) anual.	°C

Tabla 2.4. Índices de precipitación extrema

Índice	Definición	Unidades
PPt	Precipitación total en el periodo considerado.	mm
SDII	Intensidad de precipitación diaria, calculada como la cantidad de precipitación total entre el número de días húmedos (precipitación >= 1mm).	mm
CDD	Días secos consecutivos. Máxima duración de un periodo seco (precipitación diaria <1mm).	días
CDW	Días húmedos consecutivos. Máxima duración de un periodo húmedo (precipitación>=1mm).	días
R10	Número de días de precipitación superior a 10 mm.	días
R20	Número de días de precipitación superior a 20 mm.	días
R25	Número de días de precipitación superior a 25 mm.	días
RX1D	Valor máximo de precipitación diaria registrada en el periodo considerado.	mm
RX5D	Valor de precipitación máxima en un periodo de 5 días consecutivos.	mm
R90p	Precipitación en días húmedos. Número de días por encima del percentil 90 de precipitación total acumulada en el periodo considerado.	mm
R95p	Precipitación en días muy húmedos. Número de días por encima del percentil 95 de precipitación total acumulada en el periodo considerado.	mm
R99p	Precipitación en días de precipitación extrema. Número de días por encima del percentil 99 de precipitación total acumulada en el periodo considerado.	mm
R95pTot	Porcentaje de días de R95p respecto al total de días húmedos.	%

Aunque estos son los índices recomendados, no todos son útiles para el estudio del clima en Galicia por el tipo de condiciones climáticas de la región y algunos resultan redundantes sin aportar información relevante al estudio. Por todo ello, finalmente hemos considerado 10 de los 17 índices para el estudio de los extremos en temperatura y solamente 9 de los 13 índices en el de extremos de precipitación.

#### 2.2. Métodos

Para llevar a cabo el análisis de datos se utilizaron dos lenguajes de programación diferentes. Por una parte, los análisis estadísticos utilizados para determinar la posibilidad de unión de las series de datos de las estaciones de AEMET con las de Meteogalicia, junto con el cálculo de los índices de extremos climáticos, se realizó utilizando Python. En cambio, las figuras que muestran los resultados gráficamente en mapas se obtuvieron con el programa MATLAB, que cuenta con su propio lenguaje de programación.

# 2.2.1. Homogeneización de las series de temperatura y precipitación

Como ya se ha mencionado, las series de datos utilizadas fueron construidas a partir de datos de dos redes de estaciones meteorológicas diferentes. Sin embargo, aunque dichas estaciones dependen de organismos independientes, las estaciones de Meteogalicia y AEMET se encuentran ubicadas en puntos muy cercanos, por lo que se pudieron formar parejas de estaciones equivalentes. Para comprobar que es posible la unión de las series de datos de dichas estaciones es necesario comprobar que las medidas realizadas sean semejantes. Con este objetivo, se le aplicó a cada pareja un conjunto de cuatro análisis estadísticos que se indican a continuación:

 Correlación de Pearson. Permite conocer el grado de relación entre dos variables, x e y. Se calcula como el cociente entre la covarianza de los datos de las variables y el producto de la desviación estándar de cada una.

$$r_{X,Y} = \frac{cov(x,y)}{S_X S_y}$$

x = valores de referencia (AEMET)

y = valores medidos (Meteogalicia)

 $r_{X,Y}$  = Correlación de Pearson

 $\sigma_{XY}$  = covarianza de X e Y

 $S_X$  = desviación estándar de la variable X

 $S_Y$  = desviación estándar de la variable Y

Su valor oscila entre -1 y 1. Cuando  $r_{X,Y}$  = -1 o  $r_{X,Y}$  = 1, existe una correlación lineal perfecta negativa o positiva, entre las variables. En caso de que  $r_{X,Y}$  = 0, no existe relación entre las mismas (Wilks, 2006).

 Error Medio Absoluto (MAE). Mide la diferencia entre dos variables. Es el promedio aritmético del valor absoluto de la diferencia entre parejas de datos de dos variables, x e y, donde una de estas variables es la de referencia, es decir, el "valor real" y la otra el "valor medido".

Error Medio Absoluto = 
$$\frac{1}{n} \sum_{i=1}^{n} |y_i - x_i|$$

x = valores de referencia (AEMET)

y = valores medidos (Meteogalicia)

En este caso, los datos de AEMET se tomaron como "valor real", puesto que era la serie que se quería completar, y los de Meteogalicia como "valor medido".

El MAE es nulo cuando los datos de las series son completamente semejantes, por el contrario, el MAE es mayor cuanto mayores son las discrepancias entre las series (Wilks, 2006).

 Sesgo. Indica la desviación media de los datos medidos respecto a los de referencia. Se calcula como la relación entre el valor real y el observado, en este caso entre el valor medido por las estaciones de AEMET y el medido por las de Meteogalicia.

$$Sesgo = \frac{1}{n} \sum_{i=1}^{n} (y_i - x_i)$$

x = valores de referencia (AEMET)

y = valores medidos (Meteogalicia)

Si el sesgo es nulo, ambas medidas son idénticas. Un sesgo positivo indica que los valores de precipitación registrados por las estaciones de Meteogalicia son inferiores a los de referencia, mientras un sesgo negativo indica lo contrario (Wilks, 2006).

 Root Mean Square Error (RMSE). Indica el grado de dispersión de los valores medidos, en este caso los de Meteogalicia, respecto a la regresión lineal calculada a partir de los valores de referencia, los de las estaciones de AEMET.

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (y_i - x_i)^2}$$

x = valores de referencia (AEMET)

y = valores medidos (Meteogalicia)

n = tamaño de la muestra

Cuanto mayor sea la semejanza entre los valores de ambas series, menor será el valor del RMSE (Wilks, 2006).

De acuerdo con los resultados de los cuatro análisis estadísticos aplicados, de las 17 estaciones iniciales todas pueden ser completadas con los datos de su respectiva homologa excepto tres en el caso de la precipitación (Ferrol, Fornelos y Lourizán) que terminan unos años antes que el resto, como se observa en la Tabla 2.1. Se ha decidido no prescindir de ellas puesto que solamente hay una diferencia de 3 años respecto al resto de estaciones.

Una vez unidas las series, se realizó el cálculo de los índices de extremos climáticos y se procedió al estudio de la evolución y tendencia de los diferentes índices. El análisis de las tendencias, se llevó a cabo tanto anualmente como estacionalmente, ya que aunque veces en el cómputo anual no se observan tendencias significativas sí que se pueden observar en los diferentes periodos estacionales. Además, en nuestra región de estudio, localizada en las latitudes

medias del hemisferio norte, los cambios meteorológicos estacionales son muy importantes distinguiendo claramente las cuatro estaciones de primavera, verano, otoño e invierno.

La división en estaciones del año se llevó a cabo trimestralmente, de la siguiente forma:

- Invierno: diciembre, enero, febrero.
- Primavera: marzo, abril, mayo.
- Verano: junio, julio, agosto.
- Otoño: septiembre, octubre, noviembre.

En la homogeneización de las series también se buscó que estas fueran lo más completas posibles. Aunque la toma de datos mediante estaciones automáticas es buena y las series en general están completas, inevitablemente en algunos casos se encuentran huecos temporales en las series. Estos pueden ser debidos a diversos factores como posibles errores en las medidas de los instrumentos o paradas de funcionamiento de la estación por averías o reparaciones. Ha de tenerse esto en cuenta, pues dependiendo de la cantidad de datos ausentes los resultados serán más o menos representativos de la realidad. Las series se revisaron en busca de valores ilógicos (-99, -9999) que pudieran estar presentes, correspondientes a los "datos no registrados" debidos a problemas de funcionamiento o incorrecta medida de las estaciones automáticas y fueron eliminados para evitar desviaciones en los resultados. Adicionalmente, se establecieron las siguientes condiciones:

- En el cálculo de los índices de forma anual, únicamente se utilizaron años con un número de datos igual o superior a 300 días.
- En el cálculo de índices de forma estacional, solamente se utilizaron trimestres con un número máximo de 15 datos nulos.

#### 2.2.2. Tendencias

Para determinar las tendencias existentes en los índices de precipitación y temperatura extrema se ha utilizado el test de Mann-Kendall.

El motivo por el que se ha seleccionado este test en concreto es su capacidad para detectar tendencias estadísticamente significativas en series de datos que no siguen una distribución concreta (test no paramétrico) a la vez que permite la ausencia de valores en medio de las series. Otra ventaja que ofrece es la posibilidad de aplicación tanto a conjuntos de datos pequeños como grandes (Gilbert, 1987).

Para estudiar las tendencias en los resultados de los índices se utilizó Python. Las ecuaciones empleadas se muestran a continuación.

De acuerdo con Kendall, 1975 cuando el conjunto de datos es superior a 10, se calcula el test estadístico Z:

$$Z = \begin{cases} \frac{S-1}{\left[VAR(S)^{\frac{1}{2}}\right]} & si \quad S > 0 \\ 0 & si \quad S = 0 \\ \frac{S+1}{\left[VAR(S)^{\frac{1}{2}}\right]} & si \quad S < 0 \end{cases}$$

Las variables S y VAR(S) que figuran en el test Z se calculan de la siguiente forma:

$$S = \sum_{k=1}^{n-1} \sum_{j=k+1}^{n} sgn(x_j - x_k)$$

Donde n es el número de datos totales y las variables  $x_j$  y  $x_k$  cada uno de los datos de la serie (j > k). El signo de las diferencias se determina según el criterio indicado:

$$sgn(x_{j} - x_{k}) = \begin{cases} 1 & si(x_{j} - x_{k}) > 0 \\ 0 & si(x_{j} - x_{k}) = 0 \\ -1 & si(x_{j} - x_{k}) < 0 \end{cases}$$

El valor de VAR(S) se obtiene según la expresión:

$$VAR(S) = \frac{1}{18} \left[ n(n-1)(2n+5) - \sum_{p=1}^{q} t_p (t_p - 1)(2t_p + 5) \right]$$

Donde g es el número de grupos y tp el número de datos de cada grupo.

Un valor positivo de Z indica que la tendencia es un aumento, en cambio si el valor de Z es negativo indica un descenso. La significancia de las tendencias depende del p value.

- Si p value = 0.05 la tendencia es estadísticamente significativa al 95%.
- Si p value = 0.1 la tendencia es estadísticamente significativa al 90%.
- Si p value > 0.1, la tendencia no es significativa.

## 3. Resultados

Como ya se ha comentado, el análisis de los datos de temperatura y precipitación se realizó mediante el cálculo de índices extremos de precipitación y temperatura tanto de forma anual como estacional en las 17 estaciones seleccionadas. Los resultados fueron recogidos en tablas en las que las tendencias significativas se indican con dos asteriscos (\*\*) cuando esta es igual o superior al 95% (p value ≤ 0.05), y con uno (\*) cuando es igual o superior al 90% (p value ≤ 0.10). Los resultados se muestran también de forma gráfica sobre el mapa del área de estudio.

#### 3.1. Índices extremos de temperatura

En el análisis de los índices extremos de temperatura se calcularon 17 índices diferentes. Cómo se menciona en la sección de datos, estos son los índices recomendados y reconocidos por el ETCCDI. No obstante, algunos de estos índices no son realmente aplicables a la climatología de nuestra región de estudio y otros no aportan información adicional relevante con respecto a otros índices más significativos. En este trabajo finalmente solo se han incluido 10 de los 17 índices considerados como más relevantes y significativos.

#### 3.1.1. Índice TX90

Para el cálculo del índice TX90 de cada periodo (anual y estacional), se consideraron los datos diarios de temperatura máxima en el periodo de datos de cada periodo temporal. El índice TX90 representa el número de días en que la temperatura máxima excede el percentil 90 de la temperatura máxima diaria. En la Tabla 3.1 se muestran los resultados obtenidos para las once estaciones consideradas.

Tabla 3.1. Valores anuales y estacionales de la tendencia anual del índiceTX90. Los asteriscos indican las correlaciones significativas al 95% (\*\*)/90% (\*).

Estación	Anual	Invierno	Primavera	Verano	Otoño
Allariz	-0,25	-0,32**	0,22**	-0,04	-0,23*
As Pontes	0,65**	0,12**	0,25**	0,15**	0,15**
Coruña	0,65**	0,14**	0,32**	0,12**	0,18**
Coruña Aeropuerto	0,33**	0,18*	0,09*	-0,82	0,15**
Lavacolla	0,39**	0,07	0,22**	0,11**	0,12**
Lourizán	0,44**	0,20**	0,24**	0,17**	0,12**
Lugo	0,30*	0,56**	-0,28**	-0,79**	0,24**
Ourense	0,68**	0,21**	0,32**	0,31**	0,22**
Paramos Guillarei	-0,41**	0,08	0,11	-0,11	-0,11
Vigo	0,35**	0,17**	0,19**	0,12*	0,14**
Xinzo	0,21**	-0,13	0,10*	0,15**	0,11

En la Tabla 3.1 se observa una tendencia positiva generalizada del índice TX90 que se repite en prácticamente todas las localizaciones salvo en la estación de Paramos-Guillarei, en la que, aunque no se observan tendencias significativas en los periodos estacionales, se obtiene una tendencia negativa en el periodo anual. Las estaciones de Allariz y Lugo muestran variabilidad en el signo de la tendencia dependiendo del trimestre analizado.

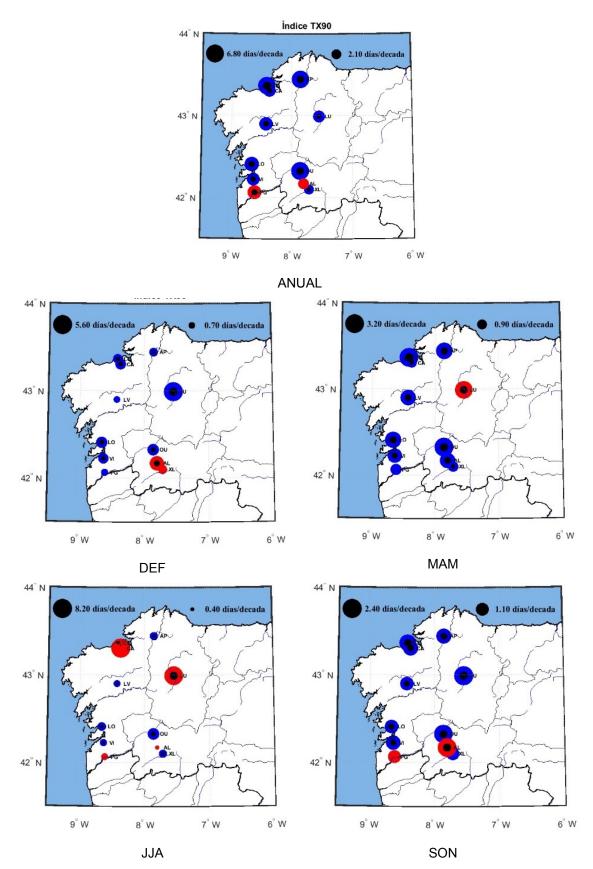


Figura 3.1. Tendencias decadales del índice TX90. Los diferentes tamaños de los círculos indican las magnitudes de las tendencias. El punto negro central señala aquellas tendencias que tienen una significancia igual o superior al 90%. Los colores rojos indican disminución del índice y los azules aumento del índice.

En la Figura 3.1 se puede ver como en el periodo anual el aumento del índice TX90 es generalizado salvo en la estación de Páramos-Gillarei, que muestra una tendencia negativa significativa, y en Allariz, donde la tendencia parece negativa, aunque no es significativa. En verano es donde se observa una mayor discordancia entre las diferentes estaciones. Lugo es la única estación que muestra una tendencia negativa y significativa en el trimestre de primavera.

#### 3.1.2. **Índice TX10**

El índice TX10 nos da información sobre días fríos, ya que nos indica el número de días/periodo en que no se supera el percentil 10 de la temperatura máxima diaria. Ese periodo puede ser anual, estacional, mensual,... Aquí analizamos el periodo anual y los cuatro periodos estacionales. En la Tabla 3.2 podemos ver las tendencias encontradas para este índice.

Tabla 3.2. Valores anuales y estacionales de la tendencia anual del índiceTX10. Los asteriscos indican las correlaciones significativas al 95% (\*\*)/90% (\*).

Estación	Anual	Invierno	Primavera	Verano	Otoño
Allariz	0,75**	0,26**	-0,01	0,01	0,20**
As Pontes	-0,46**	-0,10	-0,28**	-0,23**	-0,17**
Coruña	-0,84**	-0,15**	-0,29**	-0,30**	-0,18**
Coruña Aeropuerto	-0,48**	-1,12**	-0,06	0,05	-0,12
Lavacolla	-0,53**	-0,15**	-0,25**	-0,19**	-0,16**
Lourizán	-0,73**	-0,15**	-0,25**	-0,25**	-0,12*
Lugo	-0,54**	-0,76**	0,19**	0,48**	-0,40**
Ourense	-0,61**	-0,17**	-0,29**	-0,21**	-0,12**
Paramos Guillarei	-0,71**	-0,28**	-0,31**	0,01	-0,11*
Vigo	-0,64**	-0,19**	-0,24**	-0,19**	-0,11*
Xinzo	-0,11	0,10	-0,18**	-0,09	-0,06

Con los resultados de la Tabla 3.2, es posible afirmar que en las diferentes localizaciones repartidas por toda la geografía gallega se está produciendo un descenso generalizado del índice TX10, es decir, una disminución del número de días con temperatura máxima inferior al percentil 10. Las estaciones de Allariz y Xinzo, ambas localizadas en el suroeste de la comunidad gallega, difieren de esta tendencia al igual que Lugo en los meses de primavera y verano. La situación geográfica de las mismas puede ser una de las causas de este desacuerdo.

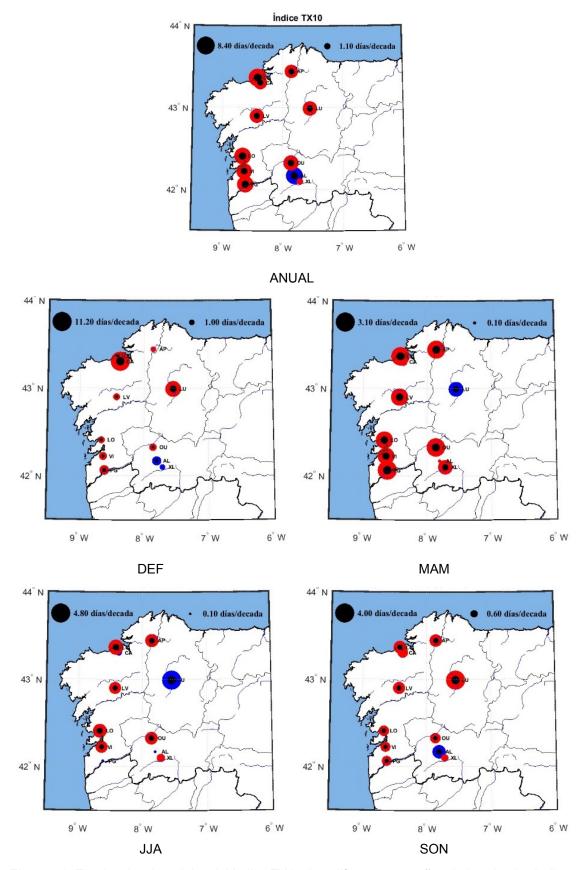


Figura 3.2. Tendencias decadales del índice TX10. Los diferentes tamaños de los círculos indican las magnitudes de las tendencias. El punto negro central señala aquellas tendencias que tienen una significancia igual o superior al 90%. Los colores rojos indican disminución del índice y los azules aumento del índice.

La Figura 3.2 nos muestra los mapas de las tendencias decadales de este índice, mostrándonos una uniformidad generalizada de las tendencias en las áreas costeras.

#### 3.1.3. Índice TN90

El índice TN90 es el número de días con temperatura mínima superior al percentil 90 de la temperatura mínima diaria. Este índice es representativo de las noches cálidas. La Tabla 3.3 muestra las tendencias encontradas para las once estaciones analizadas.

Tabla 3.3. Valores anuales y estacionales de la tendencia anual del índiceTN90. Los asteriscos indican las correlaciones significativas al 95% (\*\*)/90% (\*).

Estación	Anual	Invierno	Primavera	Verano	Otoño
Allariz	0,58**	-0,14	0,17*	0,23**	0,23
As Pontes	0,62**	0,14**	0,25**	0,29**	0,24**
Coruña	0,91**	0,12**	0,26**	0,34**	0,24**
Coruña Aeropuerto	0,68**	0,11*	0,28**	0,26**	0,23**
Lavacolla	0,58**	0,03	0,19**	0,26**	0,14**
Lourizán	0,60**	0,07*	0,18**	0,23**	0,15**
Lugo	0,29	0,03	0,14**	0,10	0,03
Ourense	0,30**	-0,10	0,17**	0,10	0,07
Paramos Guillarei	0,81**	0,08	0,41**	0,18	0,28**
Vigo	0,61**	0,04	0,23**	0,16**	0,17**
Xinzo	0,98**	0,16	0,20**	0,22**	0,31**

Las tendencias significativas obtenidas para el índice TN90 son en todos los casos positivas, lo que indica un progresivo incremento de las temperaturas mínimas anuales. En la Figura 3.3 donde se muestran las tendencias decadales observamos que ese incremento que en casi todas las localizaciones está por encima de los 3 días/década puede llegar a los 9,8 días/década. Las tendencias se muestran sobre todo en primavera y otoño que son las estaciones de transición en nuestras latitudes. Durante el invierno solo se alcanzan tendencias positivas significativas en las tres estaciones localizadas en el área del noroeste de la provincia de Coruña y en Lourizán, todas ellas localizaciones costeras.

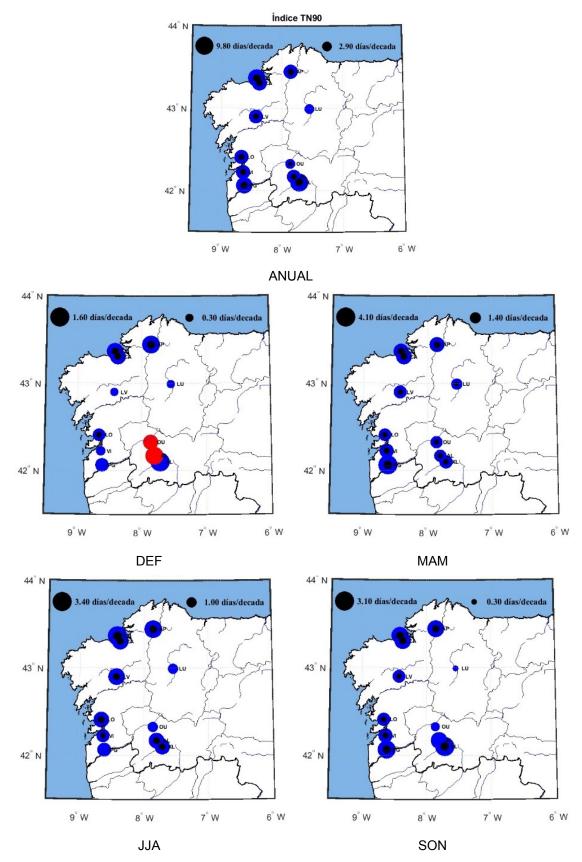


Figura 3.3. Tendencias decadales del índice TN90. Los diferentes tamaños de los círculos indican las magnitudes de las tendencias. El punto negro central señala aquellas tendencias que tienen una significancia igual o superior al 90%. Los colores rojos indican disminución del índice y los azules aumento del índice

#### 3.1.4. Índice TN10

El índice TN10 es el número de días con temperatura mínima inferior al percentil 10 de temperatura mínima diaria del periodo de referencia. Este índice se puede asociar con las noches frías y el resultado es que de forma generalizada disminuye el número de noches frías en todas las estaciones del año y consecuentemente en el recuento anual.

Tabla 3.4. Valores anuales y estacionales de la tendencia anual del índiceTN10. Los asteriscos indican las correlaciones significativas al 95% (\*\*)/90% (\*).

Estación	Anual	Invierno	Primavera	Verano	Otoño
Allariz	-0,06	0,00	-0,11	-0,19**	0,00
As Pontes	-0,49**	-0,18**	-0,28**	-0,26**	-0,26**
Coruña	-0,54**	-0,15**	-0,28**	-0,36**	-0,16**
Coruña Aeropuerto	-0,49**	-0,16	-0,26**	-0,34**	-0,14**
Lavacolla	-0,37**	-0,05	-0,20**	-0,24**	-0,15**
Lourizán	-0,19*	-0,02	-0,11*	-0,17**	-0,11*
Lugo	-0,59**	-0,21**	-0,32**	-0,24**	-0,11**
Ourense	-0,35	-0,05	-0,24*	-0,18**	-0,11
Paramos Guillarei	-0,92**	-0,28**	-0,45**	-0,21**	-0,23**
Vigo	-0,50**	-0,17**	-0,28**	-0,36**	-0,09
Xinzo	-0,46**	-0,20	-0,21**	-0,39**	-0,19**

Las tendencias recogidas en la Tabla 3.4 muestran esta disminución del índice TN10 en prácticamente todas las localizaciones consideradas salvo en Allariz y en Ourense, donde la tendencia es significativa al 95% únicamente en verano. En la Figura 3.4, donde se muestran las tendencias decadales, observamos que esta disminución que en casi todas las localizaciones está por encima de los 2 días/década en el periodo anual puede llegar a los 9,2 días/década. Las tendencias se muestran sobre todo en primavera y verano. Durante el invierno solo se alcanzan tendencias significativas en 5 de las 11 estaciones consideradas.

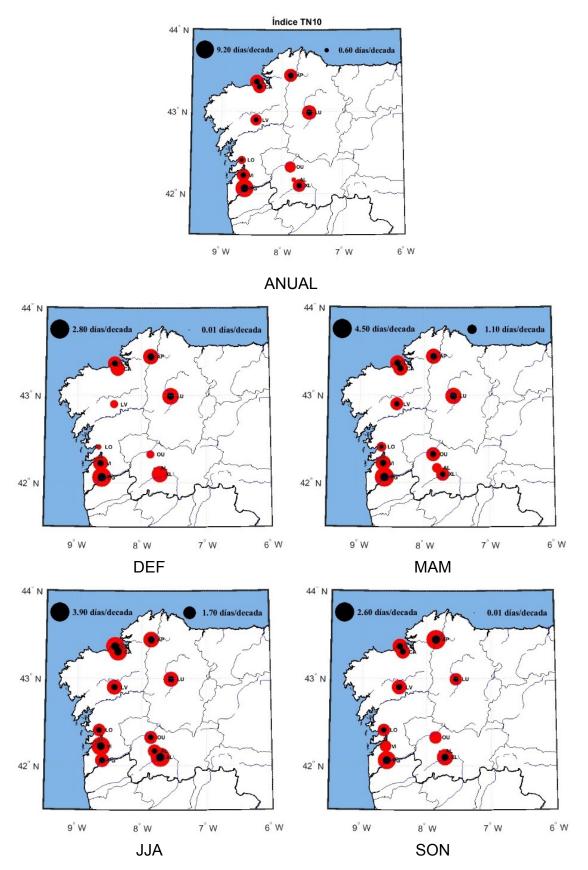


Figura 3.4. Tendencias decadales del índice TN10. Los diferentes tamaños de los círculos indican las magnitudes de las tendencias. El punto negro central señala aquellas tendencias que tienen una significancia igual o superior al 90%. Los colores rojos indican disminución del índice y los azules aumento del índice.

#### 3.1.5. Índice TXx

El índice TXx indica el valor máximo de temperatura máxima diaria en el periodo considerado. En el análisis de la Tabla 3.5 donde se presentan las tendencias anuales de este índice a lo largo del año y en los 4 periodos estacionales, no se observa una uniformidad en las mismas. Si bien parece que es en primavera y en otoño donde mayoritariamente se observan incrementos en dicho índice juntamente con el verano.

Tabla 3.5. Valores anuales y estacionales de la tendencia anual del índice TXx. Los asteriscos indican las correlaciones significativas al 95% (\*\*)/90% (\*).

Estación	Anual	Invierno	Primavera	Verano	Otoño
Allariz	-0,02	-0,11**	0,08**	-0,01	-0,08**
As Pontes	0,02	0,02	0,07**	0,05*	0,03
Coruña	0,03	-0,01	0,06**	0,05*	0,04*
Coruña Aeropuerto	0,03	0,19	0,04	-0,07	0,09**
Lavacolla	0,05**	0,01	0,06**	0,06**	0,03
Lourizán	0,04**	0,02	0,06**	0,04*	0,06**
Lugo	0,01	0,52**	-0,08**	-0,27**	0,08**
Ourense	0,10**	0,05**	0,12**	0,10**	0,13**
Paramos Guillarei	-0,04**	-0,02	0,03	-0,03	-0,04
Vigo	0,01	0,01	0,03	0,03	0,03
Xinzo	0,04**	-0,02	0,03	0,06**	0,02

Solamente en la estación de Ourense el incremento que se observa es significativo tanto anualmente como en cada una de las estaciones del año y ronda el grado de temperatura decadal salvo en invierno donde el incremento se reduce a medio grado por década.

En la figura 3.5 donde se muestran las tendencias decadales observamos que este aumento de temperatura máxima se observa de forma más generalizada en las estaciones de primavera, verano y otoño oscilando entre el medio grado y el grado por década. Llama la atención Lugo con un comportamiento contrario en primavera y verano y Allariz que parece mostrar una tendencia negativa en otoño. Durante el invierno muy pocas localizaciones muestran un incremento significativo.

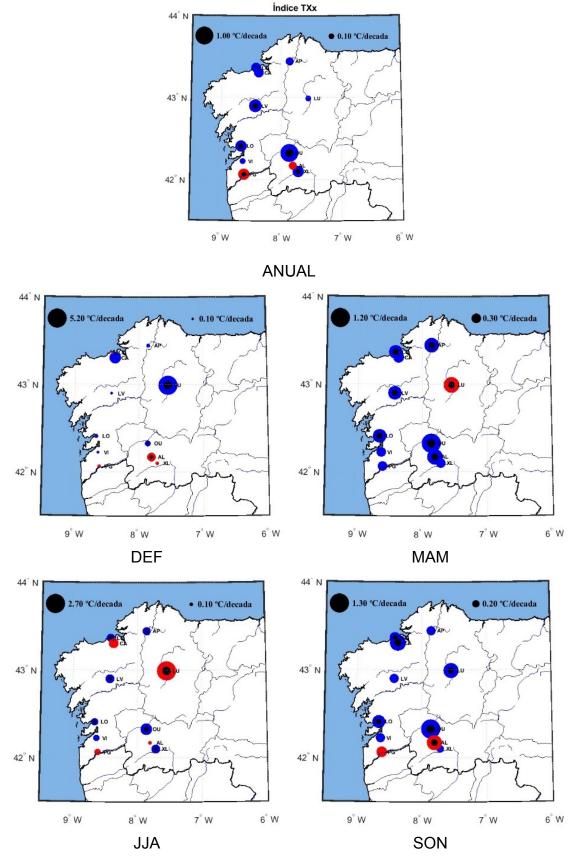


Figura 3.5. Tendencias decadales del índice TXx. Los diferentes tamaños de los círculos indican las magnitudes de las tendencias. El punto negro central señala aquellas tendencias que tienen una significancia igual o superior al 90%. Los colores rojos indican disminución del índice y los azules aumento del índice.

#### 3.1.6. Índice TNn

El índice TNn indica el valor mínimo de temperatura mínima diaria en el periodo considerado. En el análisis de la Tabla 3.6 donde se presentan las tendencias anuales de este índice a lo largo del año y en los diferentes periodos estacionales, se observa una tendencia positiva en los valores de dichas temperaturas salvo en las estaciones de Lourizán y Allariz en las que en invierno y primavera no se detecta dicho aumento de temperaturas, lo que hace que no se observe tampoco en el análisis anual. Esta tendencia no siempre es significativa pero sí que se confirma en las estaciones situadas más al oeste de la comunidad con significatividades del 95% como se puede ver en la Figura 3.6

Tabla 3.6. Valores anuales y estacionales de la tendencia anual del índiceTNn. Los asteriscos indican las correlaciones significativas al 95% (\*\*)/90% (\*).

Estación	Anual	Invierno	Primavera	Verano	Otoño
Allariz	-0,03	-0,05	-0,05	0,05**	0,00
As Pontes	0,05**	0,04*	0,04	0,06**	0,06**
Coruña	0,02	0,01	0,03	0,05**	0,03**
Coruña Aeropuerto	0,02	0,03	0,04	0,09**	0,05**
Lavacolla	0,02	0,01	0,02	0,04**	0,03*
Lourizán	-0,01	-0,01	0,01	0,03*	0,02
Lugo	0,04**	0,05**	0,07	0,04*	0,05**
Ourense	0,00	0,00	0,01	0,05**	0,06**
Paramos Guillarei	0,06**	0,06**	0,08	0,06**	0,11**
Vigo	0,04**	0,03**	0,04	0,05**	0,03
Xinzo	0,05*	0,04	0,04	0,09**	0,05**

Los meses de verano y otoño son los que experimentan un mayor aumento de los valores del índice TNn con incrementos que pueden alcanzar el 1°C/década. En primavera aunque se detectan aumentos de temperatura en dicho índice no alcanza la significatividad del 90%.

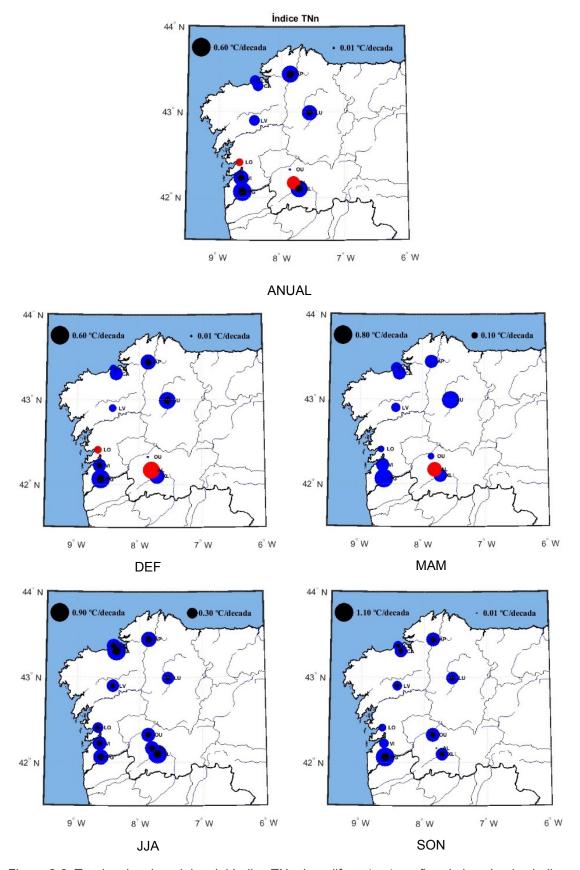


Figura 3.6. Tendencias decadales del índice TNn. Los diferentes tamaños de los círculos indican las magnitudes de las tendencias. El punto negro central señala aquellas tendencias que tienen una significancia igual o superior al 90%. Los colores rojos indican disminución del índice y los azules aumento del índice

#### 3.1.7. **Índice FD**

El FD o índice de días helados, es el número de días con temperatura mínima inferior a 0°C. Este índice presenta una disminución generalizada en toda la región de estudio y en todas las estaciones del año, salvo en verano donde apenas tienen lugar días con temperaturas por debajo de 0°C en la climatología de Galicia.

Tabla 3.7. Valores anuales y estacionales de la tendencia anual del índice FD. Los asteriscos indican las correlaciones significativas al 95% (\*\*)/90% (\*).

Estación	Anual	Invierno	Primavera	Verano	Otoño
Allariz	-0,06	-0,03	-0,11	-0,00	0,00
As Pontes	-0,51**	-0,23*	-0,18**	0,00	-0,11**
Coruña	0,00	0,00	0,00	0,00	0,00
Coruña Aeropuerto	-0,14	-0,16	-0,02	0,00	-0,03*
Lavacolla	-0,18**	-0,07	-0,06**	0,00	-0,03**
Lourizán	-0,02	0,00	0,00	0,00	-0,01
Lugo	-0,59**	-0,18	-0,25**	-0,00	-0,09**
Ourense	-0,28	-0,09	-0,07	0,00	-0,11**
Paramos Guillarei	-0,50**	-0,28**	-0,14**	0,00	-0,08**
Vigo	-0,14**	-0,10**	-0,03**	0,00	-0,01*
Xinzo	-0,83**	-0,19	-0,34**	-0,03**	-0,26**

De acuerdo con la Tabla 3.7, los principales cambios en el índice FD se observan durante la primavera y el otoño, y en menor medida durante el invierno. Si representamos las tendencias decadales podemos ver en la Figura 3.7 que la mayor disminución se observa en las localizaciones de interior rodando los 3-4 días/década en primavera y otoño y pudiendo llegar a los 8 días/década en Xinzo durante el periodo anual.

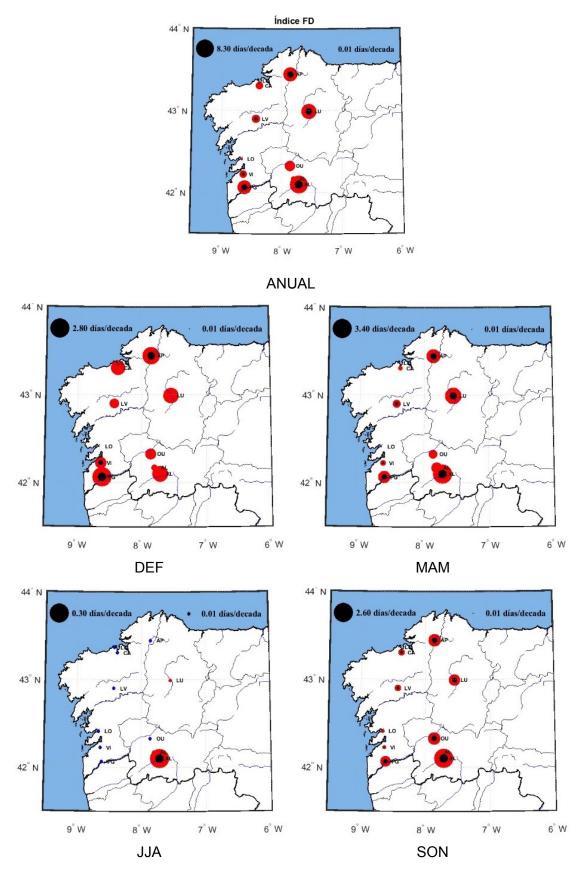


Figura 3.7. Tendencias decadales del índice FD. Los diferentes tamaños de los círculos indican las magnitudes de las tendencias. El punto negro central señala aquellas tendencias que tienen una significancia igual o superior al 90%. Los colores rojos indican disminución del índice y los azules aumento del índice

#### 3.1.8. **Índice SU**

El índice de Días cálidos (SU) es el número de días con temperatura máxima superior a 25°C. En la Tabla 3.8 se observa de forma generalizada valores positivos en las tendencias, salvo en algunas localizaciones puntuales como Allariz o Lugo que varían según la estación del año analizada.

Tabla 3.8. Valores anuales y estacionales de la tendencia anual del índice SU. Los asteriscos indican las correlaciones significativas al 95% (\*\*)/90% (\*).

Estación	Anual	Invierno	Primavera	Verano	Otoño
Allariz	-0,07	0,00	0,22**	0,22**	-0,22
As Pontes	0,62**	0,00	0,12**	0,12**	0,15**
Coruña	0,30**	0,00	0,03**	0,03**	0,13**
Coruña Aeropuerto	0,25	0,35*	0,00	0,00	0,09**
Lavacolla	0,50**	0,00	0,15**	0,15**	0,14**
Lourizán	0,58**	0,00	0,18**	0,18**	0,17**
Lugo	0,49**	1,33**	-0,08**	-0,08**	0,37**
Ourense	1,04**	0,00	0,46**	0,46**	0,33**
Paramos Guillarei	0,00	0,00	0,21**	0,21**	0,00
Vigo	0,44**	0,00	0,11**	0,11**	0,13*
Xinzo	0,44**	0,00	0,08**	0,08**	0,15*

En invierno los cambios no muestran tendencias salvo en Lugo y Coruña Aeropuerto donde se obtienen tendencias significativas al 95% y al 90%, respectivamente. Durante los trimestres de primavera, verano y otoño se observan tendencias significativas la mayoría al 95% con incrementos que rondan los 5 días/década en las localizaciones situadas más al oeste de la comunidad gallega.

En el periodo anual, esos 5 días/década pueden verse fuertemente incrementados como ocurre en la ciudad de Ourense donde se observa un aumento de 10 días/década con temperaturas por encima de los 25°C. Allariz por el contrario no muestra tendencia significativas durante el periodo anual debido a su comportamiento anómalo durante el otoño en el que se observa una disminución del índice SU que no llega a ser significativa.

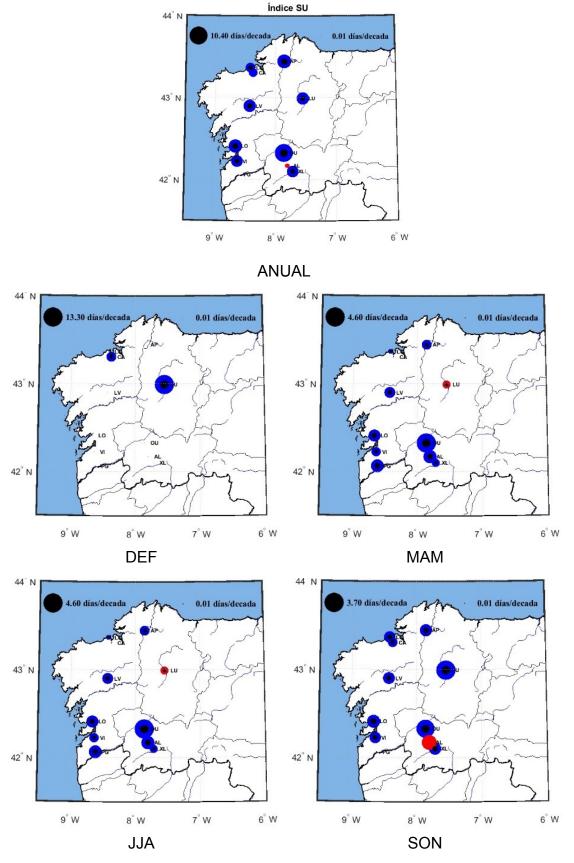


Figura 3.8. Tendencias decadales del índice SU. Los diferentes tamaños de los círculos indican las magnitudes de las tendencias. El punto negro central señala aquellas tendencias que tienen una significancia igual o superior al 90%. Los colores rojos indican disminución del índice y los azules aumento del índice.

#### 3.1.9. Índice WSDI

El índice WSDI o episodios cálidos es el número de periodos de al menos 6 días consecutivos con temperatura máxima superior al percentil 90, registrados en un periodo de tiempo determinado. El comportamiento de este índice es similar al de SU con aumentos generalizados en toda la región de estudio salvo localizaciones puntuales. En invierno y en primavera es donde estos incrementos presentan una mayor significatividad.

Tabla 3.9. Valores anuales y estacionales de la tendencia anual del índice WSDI. Los asteriscos indican las correlaciones significativas al 95% (\*\*)/90% (\*).

Estación	Anual	Invierno	Primavera	Verano	Otoño
Allariz	-0,02	-0,05**	0,01	-0,01	0,00
As Pontes	0,02**	0,01	0,02**	0,01	0,00
Coruña	0,02**	0,00	0,03**	0,01	0,01
Coruña Aeropuerto	0,01	0,02*	0,01	0,00	0,02
Lavacolla	0,02*	0,01	0,01*	0,00	0,01
Lourizán	0,02**	0,02**	0,02**	0,01	0,01
Lugo	0,00	0,05**	-0,02**	-0,03**	0,02**
Ourense	0,04**	0,01	0,02**	0,03**	0,01*
Paramos Guillarei	-0,03*	0,00	0,00	-0,02*	-0,02
Vigo	0,03**	0,01	0,01*	0,01	0,01
Xinzo	0,03**	0,01*	0,00	0,02**	0,00

Si representamos las tendencias de la Tabla 3.9 en un mapa donde se registren las tendencias por década de dicho índice se confirma que los meses de invierno y primavera son los que presentan unas tendencias más intensas. No obstante estas tendencias aunque significativas, son pequeñas sin llegar ninguna a incrementos superiores a un episodio por década.

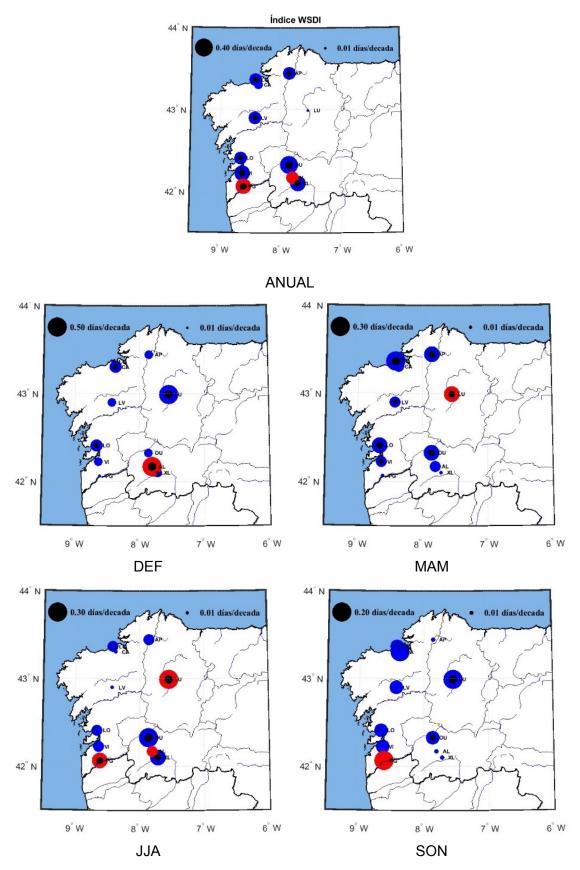


Figura 3.9. Tendencias decadales del índice WSDI. Los diferentes tamaños de los círculos indican las magnitudes de las tendencias. El punto negro central señala aquellas tendencias que tienen una significancia igual o superior al 90%. Los colores rojos indican disminución del índice y los azules aumento del índice.

#### 3.1.10. **Índice CSDI**

El índice CSDI o índice de episodios fríos es el número de periodos de al menos 6 días consecutivos con temperatura mínima inferior al percentil 10, ocurridos en un periodo de tiempo determinado. En la Tabla 3.10 se puede ver que por regla general se observa una disminución del número de dichos episodios. Esta disminución es generalizada para todos los periodos del año analizados.

Tabla 3.10. Valores anuales y estacionales de la tendencia anual del índice CSDI. Los asteriscos indican las correlaciones significativas al 95% (\*\*)/90% (\*).

Estación	Anual	Invierno	Primavera	Verano	Otoño
Allariz	-0,02	0,00	0,00	-0,03**	0,00
As Pontes	-0,02*	-0,01	-0,02**	-0,03**	-0,01
Coruña	-0,01	-0,02**	-0,01*	-0,03**	-0,01*
Coruña Aeropuerto	0,00	-0,01	-0,01	-0,04**	-0,01
Lavacolla	0,00	-0,01	-0,01	-0,03**	-0,01
Lourizán	0,00	0,00	-0,01	-0,02**	0,00
Lugo	-0,02**	-0,02**	-0,02**	-0,01	0,00
Ourense	-0,01	0,00	0,00	-0,03*	-0,01
Paramos Guillarei	-0,03**	-0,02**	-0,04**	-0,02*	-0,02**
Vigo	0,00	-0,02**	-0,02**	-0,03**	-0,01
Xinzo	-0,02	-0,01	-0,02**	-0,03**	-0,02**

Si visualizamos la Tabla 3.10 sobre un mapa se observa que los valores de las tendencias son bajas, por debajo de un episodio por década, pero significativos en la mayoría de las localizaciones sobre todo en los meses de verano y primavera. En otoño solo tres de las once localizaciones presentan significatividad en sus tendencias lo que hace que en el análisis anual también disminuya el número de localizaciones con tendencias significativas.

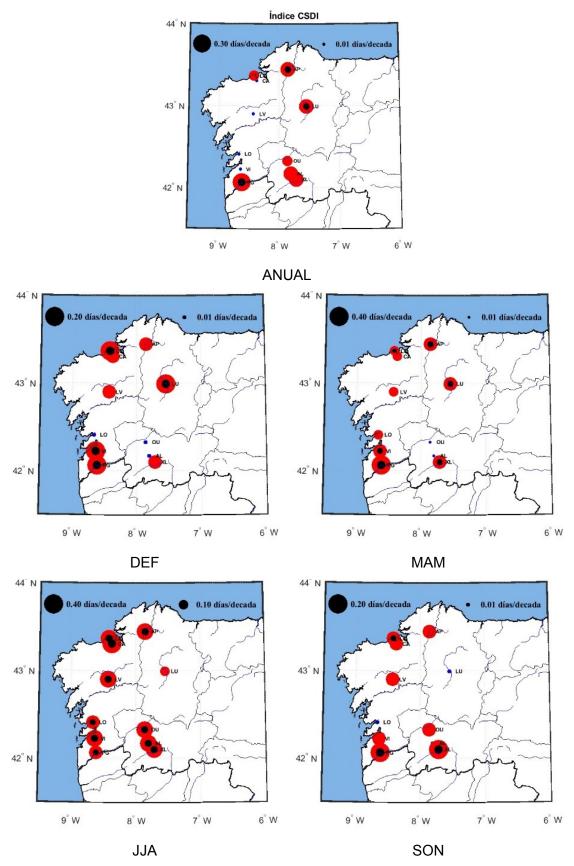


Figura 3.10. Tendencias decadales del índice CSDI. Los diferentes tamaños de los círculos indican las magnitudes de las tendencias. El punto negro central señala aquellas tendencias que tienen una significancia igual o superior al 90%. Los colores rojos indican disminución del índice y los azules aumento del índice.

## 3.2. Índices extremos de precipitación

Para el análisis de índices extremos de precipitación se calcularon 13 índices diferentes que al igual que los índices extremos de temperatura son los recomendados por el ETCCDI. Finalmente, se decidió incluir solamente los 9 índices más significativos en nuestra región de estudio que aportan los datos de mayor relevancia, quedando excluidos aquellos que no proporcionan información adicional.

## 3.2.1. Índice PrecTot

El cálculo del Índice PrecTot se realizó para los periodos anual y estacional. El índice de Precipitación Total Acumulada es la suma de la precipitación diaria de todos los días húmedos ocurridos a lo largo de un determinado periodo de tiempo, siendo días húmedos aquellos en los que la precipitación es igual o superior a 1 mm.

En la Tabla 3.11 se recogen las tendencias encontradas para este índice. Se observa una tendencia anual uniforme en todo el territorio que indica una disminución progresiva de la precipitación acumulada, aunque esta tendencia no es significativa en todas las estaciones estudiadas.

Tabla 3.11. Valores anuales y estacionales de la tendencia anual del índice PrecTot. Los asteriscos indican las correlaciones significativas al 95% (\*\*)/90% (\*).

Estación	Anual	Invierno	Primavera	Verano	Otoño
Allariz	-4,82**	-2,66**	-1,08**	-0,48	-0,44
As Pontes	-1,96	-0,91	-0,71	0,49	-0,75
Coruña	-0,62	-0,68	-0,34	0,19	0,18
Ferrol	-1,11	-1,99	0,11	0,55	0,33
Fornelos	-31,47**	-22,37*	-5,56	-1,58	2,58
Lavacolla	-11,52**	-5,72**	-3,03**	0,03	-2,36
Lugo	-2,20*	-2,23	-0,24	-0,42	0,51
Ordes	-2,55	-2,05	-0,58	0,23	-0,23
Ourense	-0,17	-1,31	0,39	-0,70	1,40
Lourizán	-3,27	-3,61	-0,95	0,66	0,71
Rozas	-0,19	2,33	1,01	-1,24	-1,21
Sarria	-12,34**	-5,07**	-2,83**	-1,22*	-2,46*
Vigo	-11,05**	-6,05**	-2,86**	-0,10	-2,08

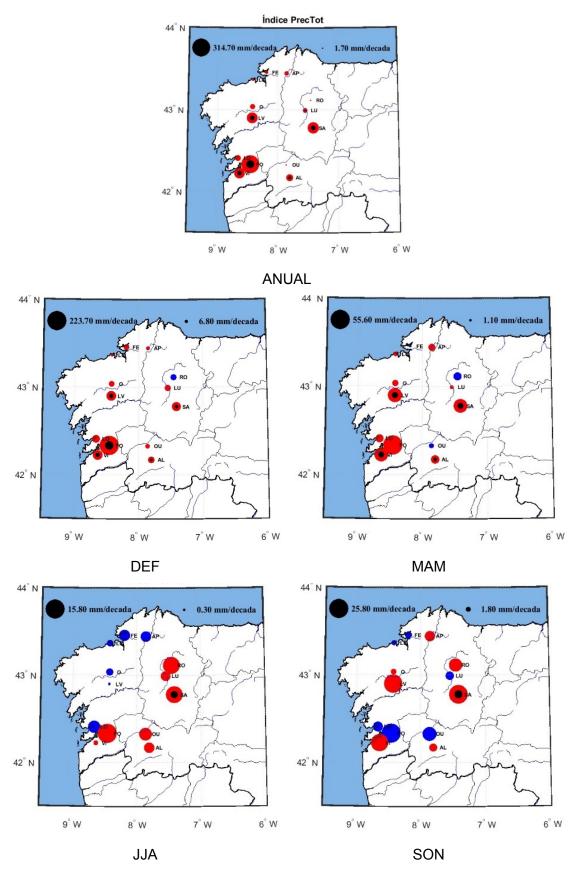


Figura 3.11. Tendencias decadales del índice PrecTot. Los diferentes tamaños de los círculos indican las magnitudes de las tendencias. El punto negro central señala aquellas tendencias que tienen una significancia igual o superior al 90%. Los colores rojos indican disminución del índice y los azules aumento del índice.

Atendiendo al análisis estacional se observa que la disminución anual de la precipitación se debe fundamentalmente a los cambios que se están produciendo durante el invierno y, en menor medida, durante la primavera. Como se puede observar en la Figura 3.11, en verano en la costa las tendencias son positivas y negativas en el interior, en otoño son variadas y en ninguno de estos trimestres son significativas. Destaca la estación de Sarria por ser la única en la que se registran tendencias significativas a lo largo de todo el año y por presentar una disminución anual de -123,4 mm/década siendo la segunda más acusada por detrás de Fornelos, que alcanza los -314,7 mm/década.

#### 3.2.2. Índice SDII

El índice SDII o Intensidad de precipitación media de los días húmedos se calcula como el sumatorio de la precipitación diaria acumulada entre el número de días húmedos (precipitación > 1 mm) de un periodo de tiempo determinado.

Comparando las figuras 3.11 y 3.12, se observa cierto paralelismo entre las tendencias anuales entre los índices PrecTot y SDII. En relación al análisis anual, se aprecia que las estaciones en las que se están produciendo descensos significativos del índice PrecTot son las mismas que en las que se observa una disminución significativa del índice SDII.

Tabla 3.12. Valores anuales y estacionales de la tendencia anual del índice SDII. Los asteriscos indican las correlaciones significativas al 95% (\*\*)/90% (\*).

Estación	Anual	Invierno	Primavera	Verano	Otoño
Allariz	-0,03**	-0,04**	-0,03**	-0,04	0,00
As Pontes	-0,02	-0,01	-0,03*	0,00	-0,03
Coruña	0,01	0,00	0,00	0,02	0,02**
Ferrol	0,00	0,00	0,02	0,03	0,02
Fornelos	-0,13**	-0,38**	-0,08	0,03	0,07
Lavacolla	-0,06**	-0,10**	-0,07**	0,00	-0,03
Lugo	-0,01	-0,04*	-0,01	0,01	0,01
Ordes	-0,02	-0,03	-0,03	0,01	0,01
Ourense	-0,01	-0,03	0,00	-0,07*	0,02
Lourizán	-0,03	-0,05*	-0,04*	0,02	0,02
Rozas	0,01	0,02	0,02	-0,02	0,00
Sarria	-0,06**	-0,07**	-0,06**	-0,01	-0,05**
Vigo	-0,07**	-0,10**	-0,07**	-0,03	-0,01

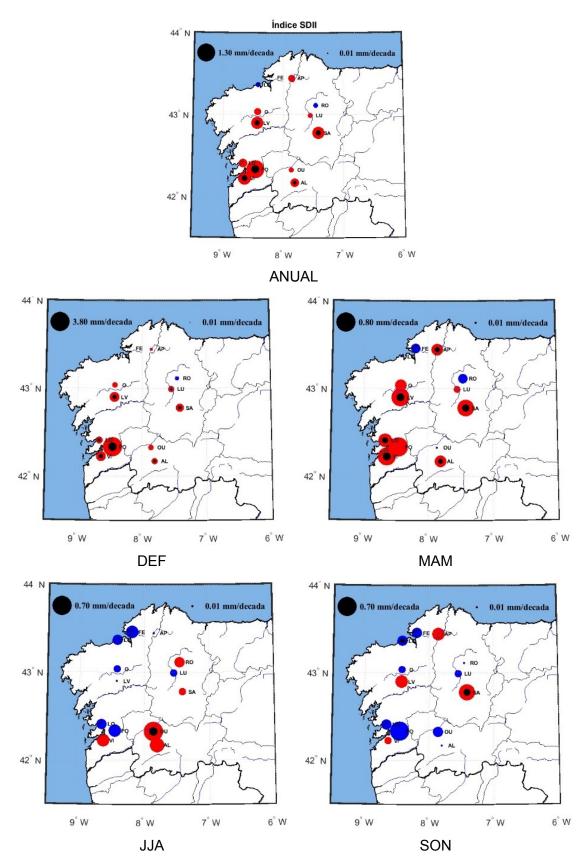


Figura 3.12. Tendencias decadales del índice SDII. Los diferentes tamaños de los círculos indican las magnitudes de las tendencias. El punto negro central señala aquellas tendencias que tienen una significancia igual o superior al 90%. Los colores rojos indican disminución del índice y los azules aumento del índice.

Los descensos del índice SDII más intensos, de hasta -3,8 mm/década, se dan en los meses de invierno. Durante los meses de verano y otoño las tendencias son variadas y no significativas en casi todos los casos, con las excepciones de Ourense en verano (-0,7 mm/década) junto con Coruña (0,2 mm/década) y Sarria (-0,5 mm/década) en otoño.

#### 3.2.3. Índice R10 mm

El Índice R10 o días de precipitación intensa, es el número de días de un determinado periodo en los que la precipitación diaria acumulada fue igual o superior a 10 mm.

Los resultados recogidos en la Tabla 3.13 indican que en general se está produciendo un descenso del número de días de precipitación intensa, aunque esta tendencia solamente es significativa en cuatro de las trece estaciones estudiadas. Los descensos más acusados del índice R10mm se corresponden con el periodo anual y alcanzan valores de hasta -5,7 días/década.

Tabla 3.13. Valores anuales y estacionales de la tendencia anual del índice R10mm. Los asteriscos indican las correlaciones significativas al 95% (\*\*)/90% (\*).

Estación	Anual	Invierno	Primavera	Verano	Otoño
Allariz	-0,19*	-0,10*	-0,02	-0,02	-0,03
As Pontes	-0,14	-0,08	-0,02	0,01	-0,04
Coruña	-0,01	-0,04	-0,01	0,01	0,03
Ferrol	0,03	-0,07	0,05	0,02	0,05
Fornelos	-0,51**	-0,25	-0,12	-0,05	0,14
Lavacolla	-0,33**	-0,18*	-0,07**	0,01	-0,07
Lugo	-0,09	-0,11	0,00	-0,01	0,01
Ordes	-0,14	-0,10	-0,01	0,00	-0,03
Ourense	-0,06	-0,09	0,02	-0,03	0,05
Lourizán	-0,12	-0,14	-0,02	0,02	0,01
Rozas	0,04	0,07	0,14	-0,06*	-0,05
Sarria	-0,57**	-0,22**	-0,14**	-0,05**	-0,15**
Vigo	-0,27**	-0,17*	-0,05	-0,01	-0,05

La ausencia de tendencias anuales significativas en la mayoría de las estaciones se debe a la variación de las tendencias en una misma ubicación a lo largo del año.

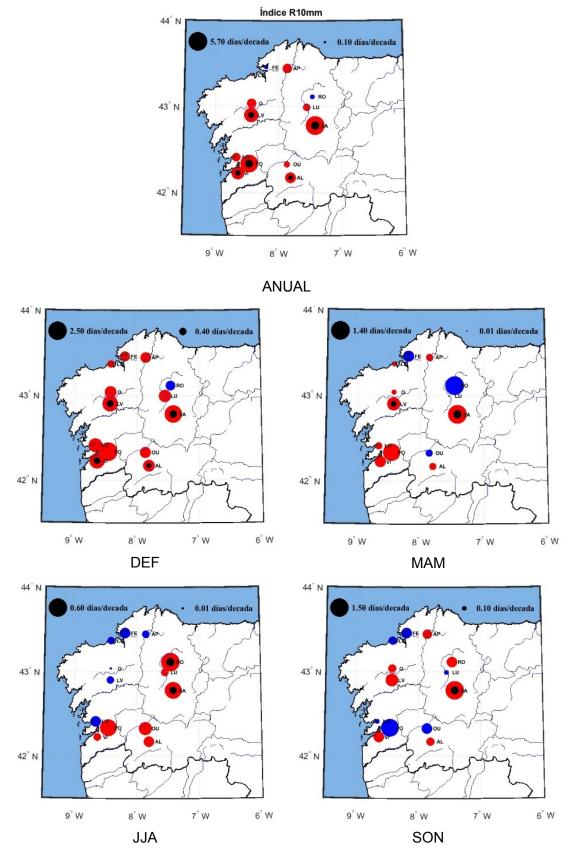


Figura 3.13. Tendencias decadales del índice R10mm. Los diferentes tamaños de los círculos indican las magnitudes de las tendencias. El punto negro central señala aquellas tendencias que tienen una significancia igual o superior al 90%. Los colores rojos indican disminución del índice y los azules aumento del índice.

Fijándonos en la Figura 3.13 podemos observar que en la mayoría de estaciones costeras se detectan tenencias negativas durante el invierno, que son positivas en los meses de verano y variables en primavera y otoño dependiendo de cada localización. Cabe destacar que en todos los casos las tendencias negativas son las únicas con una significancia igual o superior al 90%.

## 3.2.4. Índice R90p

El índice R90p nos da la cantidad de precipitación por encima de un valor umbral específico de la localización considerada para días húmedos, calculados como los percentiles 90 de la distribución de las cantidades de precipitación diaria en días con 1 mm o más de precipitación, en el período de referencia 1981–2010.

En la Tabla 3.14 se observa que la tendencia anual del R90p es negativa y significativa al 95% en la mayoría de los periodos estacionales salvo en otoño, donde no se observan tendencias significativas, y en verano, donde esas disminuciones solo son significativas en 3 de las 13 localizaciones analizadas. Las disminuciones más intensas se están produciendo en invierno, siendo el caso más destacado el de Fornelos con un valor próximo a los 147 mm/década.

Tabla 3.14. Valores anuales y estacionales de la tendencia anual del índiceR90p. Los asteriscos indican las correlaciones significativas al 95% (\*\*)/90% (\*).

Estación	Anual	Invierno	Primavera	Verano	Otoño
Allariz	-3,87**	-1,90**	-1,37**	-0,64**	-0,09
As Pontes	-0,52	-0,09	-0,82**	0,34	-0,10
Coruña	0,93	0,12	0,03	0,18	0,45
Ferrol	-0,29	-0,55	-0,10	0,52	0,22
Fornelos	-24,25**	-14,67**	-5,94	-0,29	-0,35
Lavacolla	-9,16**	-3,87**	-3,41**	0,01	-1,20
Lugo	-0,80	-1,24	-0,06	-0,30	0,51
Ordes	-1,36	-0,89	-0,91	0,26	0,22
Ourense	0,34	-0,20	0,11	-0,55**	0,97
Lourizán	-1,88	-1,29	-1,57*	0,32	1,09
Rozas	0,81	2,25	0,11	-0,61	-0,38
Sarria	-7,27**	-2,85*	-2,64**	-0,81*	-1,01
Vigo	-8,47**	-3,38**	-3,13**	-0,55	-1,00

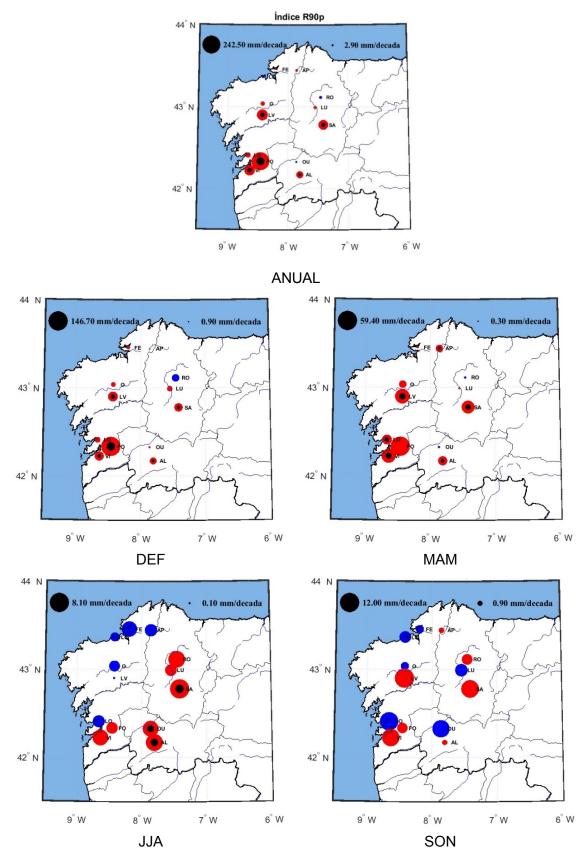


Figura 3.14. Tendencias decadales del índice R90p. Los diferentes tamaños de los círculos indican las magnitudes de las tendencias. El punto negro central señala aquellas tendencias que tienen una significancia igual o superior al 90%. Los colores rojos indican disminución del índice y los azules aumento del índice.

Representando los datos recogidos en la Tabla 3.14 sobre un mapa, se obtiene la Figura 3.14. En ella se puede observar con mayor claridad como en el análisis anual la mitad sur de la comunidad muestra disminuciones significativas que pueden llegar a los 242 mm/década. Estas tendencias se repiten en invierno y en primavera con valores algo menores. En verano las disminuciones no son tan generalizadas y a las tendencias significativas de disminución de precipitación se restringen a las tres localizaciones situadas en el suroeste de la comunidad. En otoño, tal y como vimos en la Tabla 3.14, no hay tendencias significativas obteniendo tanto valores positivos como negativos de forma indiscriminada

### 3.2.5. Índice R95pTot

El índice R95pTot es el porcentaje del número de días con precipitación acumulada superior al percentil 95 respecto al total de días húmedos registrados en un determinado periodo de tiempo.

En la Tabla 3.15 se muestran los resultados obtenidos para este índice. El periodo estacional que muestra más valores significativos es la primavera y en todas las localizaciones esa tendencia es negativa. En el resto de periodos estacionales se observan tanto valores positivos como negativos en las tendencias, siendo pocas de ellas significativas. En invierno y verano tan solo 2 localizaciones presentan disminuciones significativas y en otoño no se observa ninguna tendencia significativa.

Tabla 3.15. Valores anuales y estacionales de la tendencia anual del índice R95pTot. Los asteriscos indican las correlaciones significativas al 95% (\*\*)/90% (\*).

Estación	Anual	Invierno	Primavera	Verano	Otoño
Allariz	-0,12*	-0,16	-0,34**	-0,16	0,12
As Pontes	0,04	0,07	-0,20*	0,09	0,04
Coruña	0,07	0,03	-0,01	0,14	0,13
Ferrol	-0,03	-0,05	-0,05	0,07	0,07
Fornelos	-0,42*	-0,67*	-0,66*	-0,06	-0,16
Lavacolla	-0,18**	-0,08	-0,41**	0,03	-0,11
Lugo	0,01	-0,01	0,04	-0,28*	0,21
Ordes	-0,01	-0,01	-0,21*	0,11	0,05
Ourense	0,10	0,20	0,06	-0,50*	0,14
Lourizán	-0,04	-0,09	-0,25*	0,18	0,08
Rozas	0,13	0,67**	0,00	-0,22	0,12
Sarria	-0,21**	-0,12	-0,42**	-0,20	-0,02
Vigo	-0,21**	-0,25**	-0,47**	-0,24	-0,01

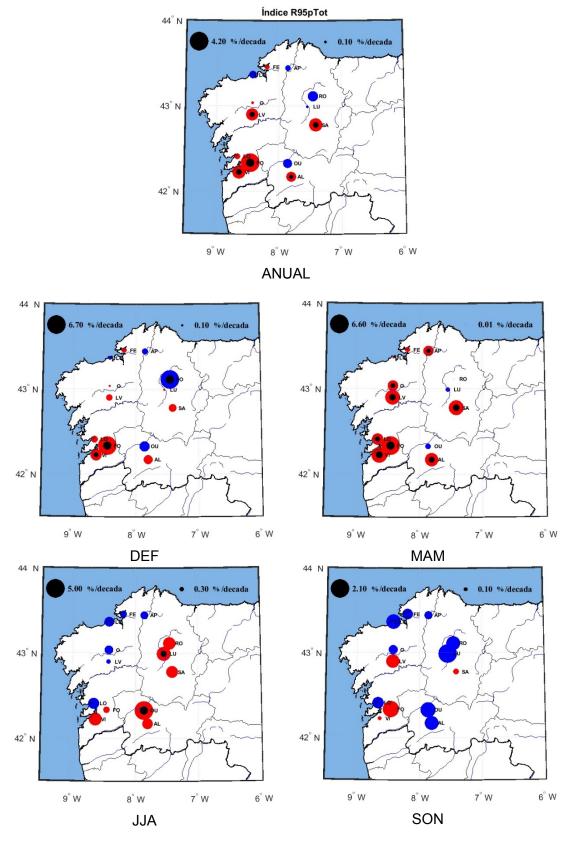


Figura 3.15. Tendencias decadales del índice R95pTot. Los diferentes tamaños de los círculos indican las magnitudes de las tendencias. El punto negro central señala aquellas tendencias que tienen una significancia igual o superior al 90%. Los colores rojos indican disminución del índice y los azules aumento del índice.

Representando los valores sobre un mapa, Figura 3.15, vemos que las disminuciones significativas se centran especialmente en la mitad sur del área de estudio, al igual que ocurría con otros índices como con el R90p. Esto se observa sobre todo en primavera y en el periodo anual. En el resto de periodos estacionales hay mayor variabilidad en los valores de las tendencias encontrándose tendencias positivas y negativas de valores similares. Incluso en otoño, aunque sin significatividad, la mayoría de los valores son positivos.

#### 3.2.6. Índice RX1D

El índice RX1D es el valor máximo de precipitación diaria registrada en un determinado periodo de tiempo en este caso en un día (anual y estacional).

Como se observa en la Tabla 3.16, las tendencias son variadas según la estación y la época del año analizada. Anualmente, son cuatro las estaciones que presentan tendencias significativas al 90% o superior, de las cuales tres son negativas (Allariz, Ferrol, Sarria) y una positiva (Lugo).

Tabla 3.16. Valores anuales y estacionales de la tendencia anual del índice RX1D. Los asteriscos indican las correlaciones significativas al 95% (\*\*)/90% (\*).

Estación	Anual	Invierno	Primavera	Verano	Otoño
Allariz	-0,15*	-0,18**	-0,24**	-0,10	-0,02
As Pontes	-0,03	0,11	-0,17	0,10	-0,18
Coruña	0,08	-0,04	0,02	0,07	0,12
Ferrol	-0,13*	-0,15	0,04	0,10	0,01
Fornelos	-0,51	-1,66**	-0,69*	0,07	0,24
Lavacolla	-0,21	-0,10	-0,36**	-0,07	-0,18
Lugo	0,19*	0,06	-0,02	-0,05	0,10
Ordes	0,18	0,18	-0,08	0,02	0,21
Ourense	0,25	0,12	-0,06	-0,16*	0,27
Lourizán	-0,05	-0,09	-0,21**	0,16	0,25
Rozas	0,16	0,40	0,05	-0,14	-0,05
Sarria	-0,45**	-0,13	-0,46**	-0,16	-0,20
Vigo	-0,24	-0,45**	-0,48**	-0,15	-0,01

En la Figura 3.16, en la que se han representado los datos recogidos por la Tabla 3.16, se observa que las tendencias significativas se dan principalmente en primavera y en invierno, aunque también hay casos como Lugo o Ferrol (-1,9 mm/década) en los que la tendencia se refleja únicamente de forma anual.

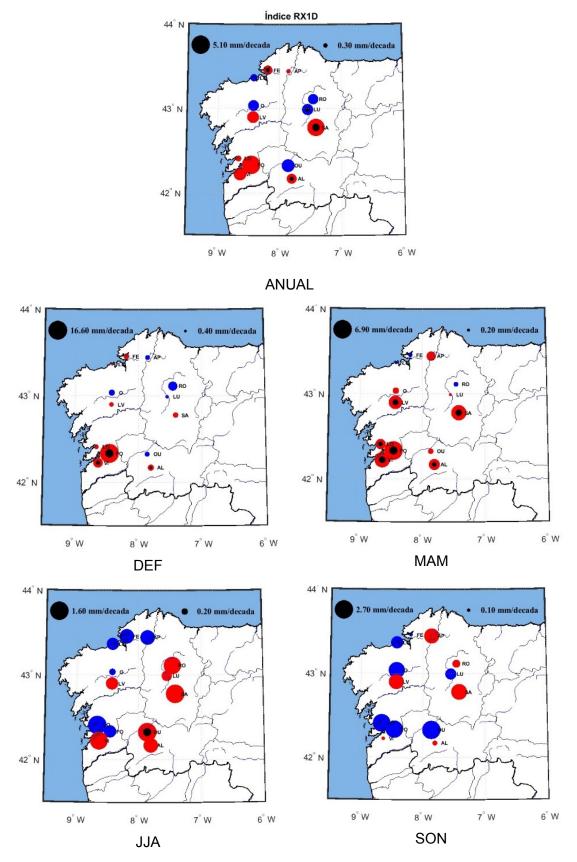


Figura 3.16. Tendencias decadales del índice RX1D. Los diferentes tamaños de los círculos indican las magnitudes de las tendencias. El punto negro central señala aquellas tendencias que tienen una significancia igual o superior al 90%. Los colores rojos indican disminución del índice y los azules aumento del índice.

#### 3.2.7. **Índice RX5D**

El índice RX5D indica el valor máximo de precipitación acumulada en un periodo de 5 días húmedos (precipitación > 1 mm) consecutivos.

Las tendencias anuales recogidas por la Tabla 3.17 son significativas al 95% en 5 de las 13 estaciones repartidas en el área de estudio e indican un descenso de la precipitación acumulada en periodos de 5 días. Se alcanzan valores anuales de hasta -36 mm/década.

Tabla 3.17. Valores anuales y estacionales de la tendencia anual del índice RX5D. Los asteriscos indican las correlaciones significativas al 95% (\*\*)/90% (\*).

Estación	Anual	Invierno	Primavera	Verano	Otoño
Allariz	-0,48**	-0,46*	-0,47**	-0,17	-0,07
As Pontes	-0,05	0,22	-0,36	0,23	-0,37
Coruña	0,16	0,21	0,19	0,13	0,00
Ferrol	-0,28	-0,23	-0,09	0,20	-0,04
Fornelos	-3,60**	-4,38**	-0,72	-0,22	-0,40
Lavacolla	-1,04**	-0,53	-0,81**	-0,05	-0,68
Lugo	0,03	0,00	-0,09	-0,12	0,07
Ordes	-0,03	0,05	-0,27	0,18	0,13
Ourense	-0,11	-0,09	-0,02	-0,25	0,22
Lourizán	-0,33	-0,36	-0,38	0,15*	0,20
Rozas	0,08	1,08*	0,11	0,13	-0,57
Sarria	-0,88**	-0,52	-0,84**	-0,41**	-0,47*
Vigo	-1,04**	-0,84**	-1,01**	-0,24	-0,22

Representando los resultados del índice RX5D sobre un mapa se obtiene la Figura 3.17 en la que destacan los meses de invierno a primavera, pues es en este periodo donde mayor número de tendencias son significativas al 90% y 95% y se registran disminuciones de hasta -43,8 mm/década. En verano se observan tendencias positivas en las localizaciones situadas más al norte y oeste de la comunidad, aunque solamente es significativa la de Lourizán que presenta un ascenso de 1,5 mm/década. Las localizaciones situadas en la mitad sur y este por el contrario presentan valores negativos que llegan a ser significativos en Sarria con -4,1 mm/década. En otoño predominan los valores negativos aunque la única tendencia significativa es la de Sarria (-4,7 mm/década).

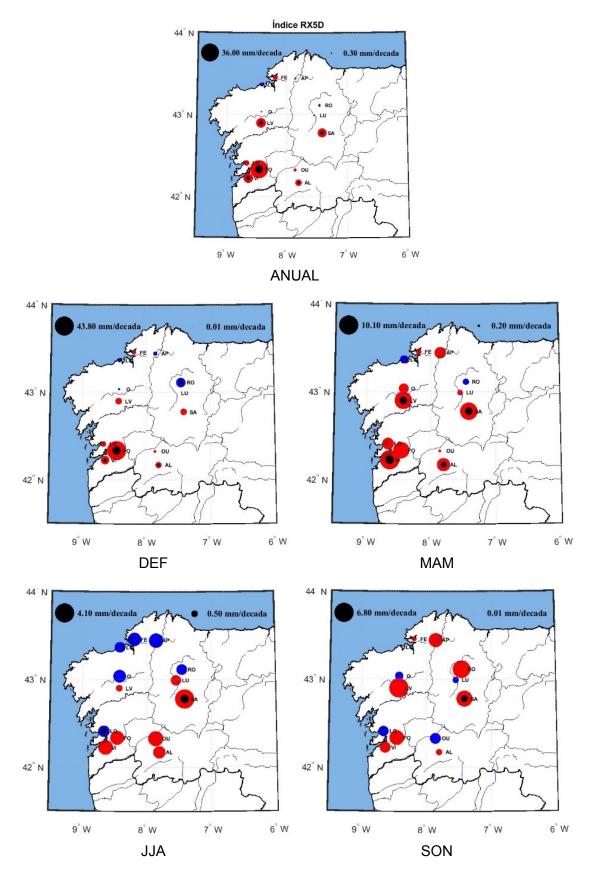


Figura 3.17. Tendencias decadales del índice RX5D. Los diferentes tamaños de los círculos indican las magnitudes de las tendencias. El punto negro central señala aquellas tendencias que tienen una significancia igual o superior al 90%. Los colores rojos indican disminución del índice y los azules aumento del índice.

#### 3.2.8. Índice CDD

El Índice CDD es una medida de la duración de los periodos de días secos consecutivos registrados a lo largo de un periodo de tiempo determinado.

En el estudio anual Tabla 3.18 se observa una mayoría de valores negativos en las tendencias del índice CDD aunque sólo es significativa la de As Pontes y Ordes. En contraposición Sarria presenta una tendencia positiva y significativa al 95%.

Tabla 3.18. Valores anuales y estacionales de la tendencia anual del índice CDD. Los asteriscos indican las correlaciones significativas al 95% (\*\*)/90% (\*).

Estación	Anual	Invierno	Primavera	Verano	Otoño
Allariz	-0,06	-0,03	-0,05	0,00	0,04
As Pontes	-0,12*	0,01	-0,01	-0,13*	-0,02
Coruña	0,001	-0,06	0,02	-0,01	0,05*
Ferrol	-0,07	-0,03	-0,04	-0,01	0,01
Fornelos	-0,13	0,13	0,05	-0,10	-0,12
Lavacolla	0,001	-0,04	0,01	-0,04	0,05
Lugo	-0,16	-0,06	0,13**	-0,09	0,03
Ordes	-0,04*	0,01	0,00	-0,08**	0,01
Ourense	0,001	-0,07	0,13	0,34**	0,02
Lourizán	-0,05	0,00	-0,02	-0,11	-0,05
Rozas	0,01	0,01	-0,03	0,20	0,02
Sarria	0,22**	0,02	0,05	0,16	0,06
Vigo	-0,02	-0,02	0,01	0,01	0,01

Esta homogeneidad negativa anual no se mantiene en el estudio de los diferentes periodos estacionales. Como se observa en la Figura 3.18, durante los meses de verano la tendencia predominante en las estaciones de costa es negativa, es decir, en estas ubicaciones los periodos de días secos tienden a acortarse, en cambio en las estaciones de interior ocurre lo contrario y en localizaciones como Ourense se registra un incremento de 3,4 días/década. En general, durante el resto del año las tendencias varían entre positivas y negativas pero sin llegar a ser significativas. Las excepciones son Lugo en primavera (1,3 días/década) y A Coruña en otoño (0,5 días/década). En otoño parece predominar el aumento del índice CDD.

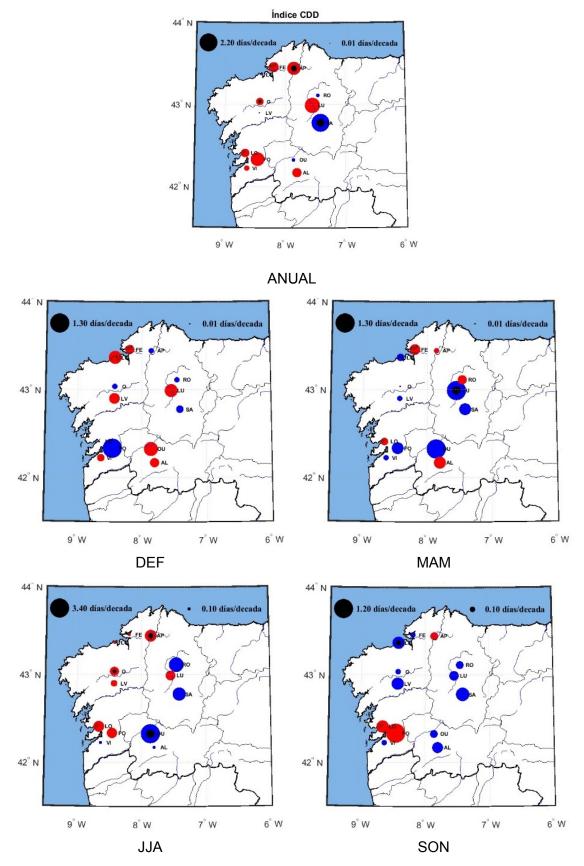


Figura 3.18. Tendencias decadales del índice CDD. Los diferentes tamaños de los círculos indican las magnitudes de las tendencias. El punto negro central señala aquellas tendencias que tienen una significancia igual o superior al 90%. Los colores rojos indican disminución del índice y los azules aumento del índice.

#### 3.2.9. Índice CWD

El Índice CWD es una medida de la duración de los periodos de días húmedos consecutivos, registrados a lo largo de un periodo de tiempo determinado.

De acuerdo con la Tabla 3.19, las tendencias son en general negativas para todo el territorio, aunque en pocos casos son significativas. Los descensos anuales oscilan alrededor de 0,5 días/década, aunque en algún caso muestran una reducción de hasta 1,5 días/década como se observa en Fornelos. En el estudio estacional las tendencias mantienen los valores negativos pero sólo Ordes en invierno y Fornelos en verano presentan valores significativos.

Tabla 3.19. Valores anuales y estacionales de la tendencia anual del índice CWD. Los asteriscos indican las correlaciones significativas al 95% (\*\*)/90% (\*).

Estación	Anual	Invierno	Primavera	Verano	Otoño
Allariz	-0,04*	-0,03	-0,05	0,00	-0,01
As Pontes	0,01	0,01	-0,03	0,00	-0,02
Coruña	-0,01	0,01	-0,02	0,00	-0,03
Ferrol	-0,06	-0,07	-0,03	-0,01	-0,06
Fornelos	-0,15*	-0,04	-0,06	-0,07**	-0,05
Lavacolla	-0,05**	-0,03	-0,01	0,01	-0,06
Lugo	-0,02	-0,01	-0,02	-0,01	-0,01
Ordes	-0,01	-0,03*	0,01	0,00	-0,03
Ourense	0,01	0,01	-0,01	-0,01	0,01
Lourizán	-0,03	-0,06	0,00	0,00	-0,02
Rozas	-0,01	0,05	-0,02	-0,04	-0,07
Sarria	-0,05*	-0,06	-0,01	-0,02	-0,05
Vigo	-0,05*	-0,04	-0,03	0,00	-0,03

Si representamos en un mapa, Figura 3.19, los valores de la tabla se observa una homogeneidad hacia una disminución de la longitud del periodo de días consecutivos húmedos en las estaciones de primavera, verano y otoño. En invierno esa homogeneidad no es tan clara, sin embargo, ninguna tendencia positiva llega a ser significativa.

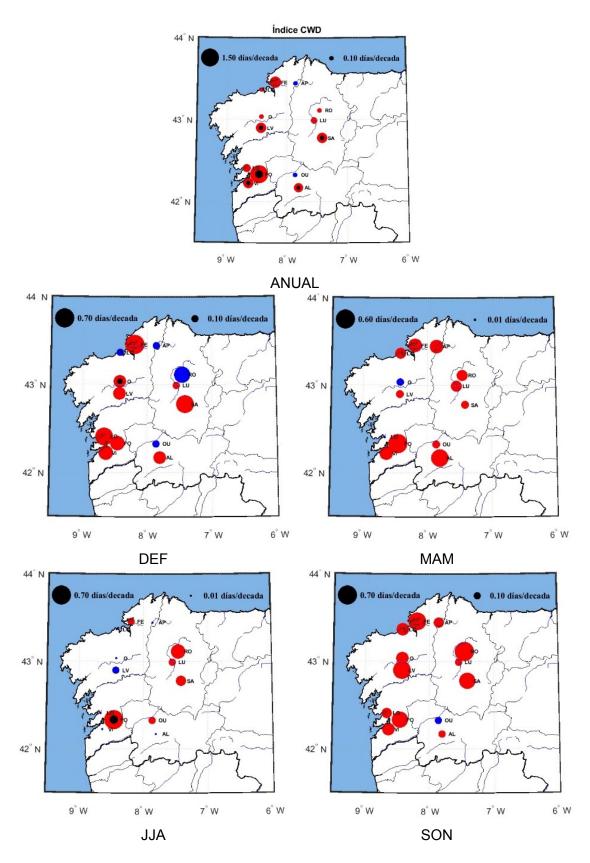


Figura 3.19. Tendencias decadales del índice CWD. Los diferentes tamaños de los círculos indican las magnitudes de las tendencias. El punto negro central señala aquellas tendencias que tienen una significancia igual o superior al 90%. Los colores rojos indican disminución del índice y los azules aumento del índice.

# 4. Discusión y conclusiones

A partir de los resultados de índices extremos de precipitación y temperatura obtenidos en este trabajo y teniendo en cuenta los resultados de otros estudios previos, se ha llegado a una serie de conclusiones que se exponen a continuación.

# 4.1. Índices de temperatura

Cada uno de los índices calculados aporta información sobre algún aspecto aislado del comportamiento de los extremos de temperatura, tanto en relación a las temperaturas máximas como a las mínimas. Teniendo en cuenta todos los resultados obtenidos se observa que en general, en Galicia, tanto las temperaturas máximas como las mínimas muestran aumentos con tendencias significativas generalizadas.

Coincidiendo con las conclusiones de estudios previos realizados tanto para la Comunidad Autónoma de Galicia (Cruz et al., 2009) como para el conjunto de la Península Ibérica (Rodríguez-Puebla et al., 2010, Andrade et al., 2012), las tendencias observadas en el número de días cálidos (TX90) y fríos (TX10) indican un incremento de los días cálidos (TX90) a lo largo del año, a la vez que

disminuye el número de días fríos (TX10). Además, estas tendencias sugieren que la reducción de la frecuencia de los días fríos es más intensa que el aumento de los cálidos, un resultado semejante fue encontrado por estudios previos como el de Cruz et al.,2009 realizado a nivel regional, y por Brunet et al, 2007, a nivel peninsular, aunque es contrario a lo encontrado a nivel europeo por Tank y Können, 2003, ya que a esta escala se concluye que el incremento de las temperaturas es debido fundamentalmente al incremento de los días cálidos en lugar de a la reducción de los fríos.

En cuanto al número de noches cálidas (TN90) la tendencia es positiva y uniforme en todo el territorio, ocurre lo contrario con el número de noches frías (TN10) que muestra tendencia negativa. Se trata en ambos casos de tendencias observadas a lo largo del año. La reducción del número de noches frías (TN10) es más intensa que la de días fríos (TX10) y el aumento de noches cálidas (TN90) mayor que el de días cálidos (TX90). Además, el comportamiento de las tendencias para los índices relacionados con las temperaturas mínimas (TN90, TN10) es más homogéneo que el de los índices relacionados con las temperaturas máximas.

En el estudio por estaciones de estos índices se observa que el incremento de estos índices se está produciendo especialmente en los meses de primavera y otoño, esto parece indicar que las estaciones de transición tienden a ser más cálidas de forma que se estaría produciendo una ampliación de la duración del verano.

El comportamiento de los índices TXx y TNn es similar al de los anteriores. Las tendencias en los valores máximos de temperatura máxima (TXx) son positivas y especialmente significativas en los meses de primavera, otoño y en menor medida en los de verano. El valor mínimo de temperatura mínima (TNn) indica también una tendencia positiva especialmente homogénea en los meses de verano y otoño.

El índice de número de días con temperaturas mínimas por debajo de 0°C (FD) concuerda con el resto de índices que indican un incremento de la temperatura mínima, ya que en este caso la tendencia es negativa. El descenso en los días con helada aparece reflejado en las tendencias estacionales excepto en verano,

donde no existen tendencias significativas debido a la escasa ocurrencia de este fenómeno durante el periodo veraniego.

La tendencia anual en el número de días de verano SU, muestra un aumento generalizado en el número de días con temperaturas máximas superiores a 25°C en toda la región, aunque si observamos el estudio estacional se ve que en invierno las tendencias no llegan a ser significativas, con la excepción de dos ubicaciones situadas al norte de la región de estudio. Estos resultados concuerdan con los encontrados en el estudio de Fernández-Montes y Rodrigo, 2012 donde también se muestra que en las regiones costeras de la Península, entre las que se incluye nuestra región de estudio, existe una tendencia negativa en el índice FD y positiva en el índice SU.

El índice WSDI, que indica el número de periodos de 6 o más días de duración con temperaturas máximas superiores al percentil 90, muestra una tendencia anual positiva y significativa en casi toda la región que se debe principalmente a las tendencias encontradas en los meses de invierno y primavera. El índice CSDI, relacionado con periodos donde las temperaturas mínimas se encuentran por debajo del percentil 90 durante al menos seis días consecutivos, presenta mayoritariamente tendencias negativas que alcanzan valores significativos casi generalizados en los meses de primavera y verano. Esto coincide con lo observado en estudios previos realizados en Portugal (Ramos et al., 2011), región colindante a Galicia. Si bien en ambos casos los incrementos o disminuciones, según el índice de que hablemos, WSDI o CSDI, son pequeños y no alcanzan el episodio por década, las tendencias se muestran homogéneas y alcanzan la significatividad en muchas de las localizaciones consideradas reafirmando la tendencia a un aumento de temperatura en la región.

### 4.2. Índices de precipitación

Las tendencias encontradas en este estudio en cuanto a la cantidad de precipitación total acumulada (PrecTot) son negativas en todo el territorio. A diferencia de los resultados obtenidos en el estudio de Cruz et al., 2009, realizado en la misma área de estudio con series de datos que finalizaban en el

año 2006, en el que las tendencias no eran significativas, en nuestro análisis cinco de las trece estaciones utilizadas, situadas en el centro y sur de Galicia, muestran tendencias significativas decrecientes que se repiten en invierno y primavera. Destaca el caso de Sarria, donde la tendencia negativa es significativa en los cuatro periodos estacionales, y el de la estación de Fornelos donde se observa una tendencia de -315 mm/década.

El índice SDII, utilizado para analizar los cambios en la concentración de la precipitación y que es el cociente obtenido de dividir la precipitación total anual entre el número de días con precipitación, presenta un comportamiento similar al de PrecTot. La intensidad de la precipitación (SDII) muestra mayoritariamente tendencias negativas que son significativas anualmente debido a los valores obtenidos en los meses de invierno y primavera en las cinco estaciones situadas en el interior y sur de la región. Las estaciones situadas al norte presentan una mayor variabilidad en las tendencias y en ningún caso llegan a ser significativas. En trabajos como el de de Lima et al., 2014, que realizó el cálculo de este índice para la región de Portugal en el periodo 1941-2007, la conclusión en cuanto al comportamiento de la intensidad de precipitación fue similar a la encontrada en este estudio.

Los resultados obtenidos para los índices, RX1D, RX5D, R90p y R95pTot presentan un comportamiento semejante. Tendencias negativas y significativas debidas sobre todo a lo que ocurre en los meses de primavera en las estaciones del interior y sur de la comunidad y tendencias variables y no significativas en su mayor parte en la región norte de Galicia. El índice R10mm refleja una disminución generalizada en el territorio del número de días de precipitación superior a 10 mm, aunque dichas tendencias de nuevo solo se muestran significativas en las zonas del interior y sur de Galicia.

El índice que registra la duración de los periodos secos (CDD) muestra una gran variabilidad en las tendencias asociada tanto al periodo del año analizado como a la localización geográfica del punto analizado. Anualmente, la tendencia dominante en las estaciones de costa es negativa alcanzando valores significativos únicamente en As Pontes con descenso de 1,2 días/década, mientras que estaciones de interior como Ourense, Rozas o Sarria presentan una tendencia positiva que es significativa únicamente en Sarria donde se

observa un aumento de 2,2 días/década. El invierno repite este patrón de comportamiento y en el resto de periodos estacionales la variabilidad es tan alta que es difícil extraer una conclusión clara. La tendencia en el índice que registra la longitud de los periodos húmedos (CWD) es negativa, mostrándose uniforme en todo el territorio y a lo largo del año, aunque los valores significativos se obtienen principalmente en el análisis anual de las cinco estaciones del interior y sur de Galicia.

En general, los resultados de este trabajo indican que las tendencias existentes en los índices de precipitación se inclinan hacia un descenso de los mismos en casi todos los casos, aunque por ahora son descensos relativamente pequeños. Además, los valores varían bastante según la localización geográfica y la época del año. Esto nos lleva a decir que los cambios aún no son muy grandes, pero parecen decantarse hacia disminuciones encontrándose significatividades que estudios realizados aproximadamente una década atrás no mostraban (Cruz et al., 2009).

Una posible continuación de este trabajo podría consistir en el cálculo de otros índices relativos a los eventos de precipitación extrema con una mayor sensibilidad a los cambios en la torrencialidad de la precipitación. Un posible índice a estudiar sería el índice de concentración de precipitación (CI), que es capaz de relacionar los eventos de precipitación con la escala temporal y espacial en la que ocurren y que resulta de especial interés por su aplicación para conocer el grado de erosión provocado por la precipitación (Serrano-Notivoli et al., 2017). Aunque la región noroeste de la Península Ibérica presenta valores bajos del índice CI en relación con el resto del territorio (Martín-Vide, 2004), en estudios más recientes (Serrano-Notivoli et al., 2017; de Luis et al., 2011) se encontró que, aunque con variaciones espaciales y temporales en función del periodo estacional analizado, anualmente se observa una tendencia creciente de dicho índice en todo el territorio de España.

## REFERENCIAS

- Andrade, C., Leite, S.M., Santos, J.A. (2012). Temperature extremes in Europe: overview of their driving atmospheric patterns. Natural Hazards and Earth System Sciences, 12:1671-1691. DOI: 10.5194/nhess-12-1671-2012
- Brunet, M., Jones, P.D., Sigró, J., Saladié, O., Aguilar, E., Moberg, A., Della-Marta, P.M., Lister, D., Walther, A., López, D. (2007), Temporal and spatial temperature variability and change over Spain during 1850–2005. Journal of Geophysical Research: Atmospheres, 112. DOI:10.1029/2006JD008249
- Cruz, R., Lago, A., Lage, A., Rial, M.E., Diaz-Fierros, F., Salsón, S. (2009). Evolución recente do clima de Galicia. Tendencias observadas nas variables meteorolóxicas. En Xunta de Galicia Consellería de Medio Ambiente e Desenvolvemento Sostible (Ed.), Evidencias e Impacto do Cambio Climático en Galicia (pg. 19-58). Tórculo Artes Gráficas S.A.
- de Lima, M.I.P., Trigo, R.M., Santo, F.E., Ramos, A.M. (2014). Trends and correlations in annual extreme precipitation indices for mainland Portugal,

- 1941–2007. Theorical and Applied Climatology, 119:55-75. DOI: 10.1007/s00704-013-1079-6
- de Luis, M., Gonzalez-Hidalgo, J.C., Brunetti, M., Longares, L.A. (2011) Precipitation concentration changes in Spain 1946-2005. Natural Hazards and Earth System Sciences, 11:1259-1265. DOI: 10.5194/nhess-11-1259-2011.
- Fernández-Montes, S., Rodrigo, F.S. (2012). Trends in seasonal indices of daily temperature extremes in the Iberian Peninsula, 1929–2005. International Journal of Climatology, 32:2320-2332. DOI: 10.1002/joc.3399
- Gilbert, R.O. (1987). Statistical Methods for Environmental Pollution Monitoring. Estados Unidos: Van Nostrand Reinhold Company.
- Gómez-Gesteira, M., Gimeno, L., deCastro, M., Lorenzo, M.N., Alvarez, I., Nieto, R., Taboada, J.J., Crespo, A.J.C., Ramos, A.M., Iglesias, I., Gomez-Gesteira, J.L., Santo, F.E., Barriopedro, D., Trigo, I.F. (2011). Th estate of climate in NW Iberia. Climate Research, 48:109-144. DOI: 10.3354/cr00967.
- IPCC, Intergovernmental Panel on Climate Change (2013). Climate Change 2013. The Physical Science Basis. Working Group I Contribution to the Fifth Assessment Report of the Intergovernmental Panel on Climate Change. Cambridge University Press.
- Kendall, M.G. (1975). Rank Correlation Methods. 4th edition. Londres:
   Charles Griffin.
- Martín-Vide, J. (2004). Spatial Distribution of a Daily Precipitation
   Concentration Index in Peninsular Spain. International Journal of Climatology,
   24:959-971. DOI: 10.1002/joc.1030.
- Martínez-Cortizas, A., Pérez-Alberti, A. (1999). Atlas Climático de Galicia.
   Santiago de Compostela: GRAFICANT S.L.

- Naranjo, L., Pérez-Muñuzuri, V. (2006). A Variabilidade natural do clima en Galicia. Edita Consellería de Medio Ambiente e Desenvolvemento Sostible, Xunta de Galicia. ISBN 978-84-96494-74-8
- Peterson, T.C. (2005). Climate Change Indices. World Meteorological Organization Bulletin, 54, 2.
- Ramos, A.M., Trigo, R.M., Santo, F.E. (2011). Evolution of extreme temperatures over Portugal: recent changes and future scenarios. Climate Research, 48:177-192. DOI: 10.3354/cr00934.
- Rodrigo, F.S., Trigo, R.M. (2007). Trends in daily rainfall in the Iberian Peninsula from 1951 to 2002. International Journal of Climatology, 27:513-529. DOI: 10.1002/joc.1409.
- Rodríguez-Puebla, C., Encinas, A.H., García-Casado, L.A., Nieto, S. (2010).
   Trends in warm days and cold nights over the Iberian Peninsula: relationships to large-scale variables. Climatic Change, 100:667–684. DOI: 10.1007/s10584-009-9721-0
- Santo, F.E., deLima, M.I.P., Ramos, A.M., Trigo, R.M. (2014). Trends in seasonal surface air temperature in mainland Portugal, since 1941. Royal Meteorological Society, 34:1814-1837. DOI: 10.1002/joc.3803.
- Serrano-Notivoli, R., Martín-Vide, J., Saz, M.A., Longares, L.A., Begería, S., Sarricolea, P., Meseguer-Ruiz, O., de Luis, M. (2017) Spatio-temporal variability of daily precipitation concentration in Spain based on a high-resolution gridded data set. International Journal of Climatology, 38 (Suppl.1): e518-e530. DOI: 10.1002/joc.5387.
- Tank, M.G.K., Können, G.P. (2003). Trends in Indices of Daily Temperature and Precipitation Extremes in Europe, 1946–99. Royal Netherlands

- Meteorological Institute, De Bilt, Netherlands. DOI: 10.1175/1520-0442(2003)016<3665:TIIODT>2.0.CO;2
- Wilks, D.S. (2006). Statistical Methods in the Atmospheric Sciences. Estados Unidos: ELSEVIER.
- Yan, Z., Jones, P.D., Davies, T.D., Moberg, A., Bergstrom, H., Camuffo, D., Cocheo, C., Maugeri, M., Demarée, G.R., Verhoeve, T., Thoen, E., Barriendos, M., Rodriguez, R., Martín-Vide, J., Yang, C. (2002). Trends of extreme temperatura in Europe and China base don daily observations. Climatic Change, 53:355-392. DOI: 10.1007/978-94-010-0371-1 13.
- Zhang, X., Alexander, L., Hegerl, G.C., Jones, P., Tank, A.K., Peterson, T.C., Trewin, B., Zwiers, F.W. (2011). Indices for monitoring changes in extremes based on daily temperature and precipitation data. WIREs Clim Change, 2:851-870. DOI: 10.1002/wcc.147.

### PÁGINAS WEB CONSULTADAS

- AEMET. <a href="http://www.aemet.es/es/portada">http://www.aemet.es/es/portada</a>]
- European Environmental Agency (2017). Mean precipitation.
   <a href="https://www.eea.europa.eu/downloads/03c36dec1c084253989eb5196be51b">https://www.eea.europa.eu/downloads/03c36dec1c084253989eb5196be51b</a>
   <a href="https://www.eea.europa.eu/downloads/03c36dec1c084253989eb5196be51b">https://www.eea.europa.eu/downloads/03c36dec1c084253989eb5196be51b</a>
   <a href="https://www.eea.europa.eu/downloads/03c36dec1c084253989eb5196be51b">https://www.eea.europa.eu/downloads/03c36dec1c084253989eb5196be51b</a>
   <a href="https://www.eea.europa.eu/downloads/03c36dec1c084253989eb5196be51b">https://www.eea.europa.eu/downloads/03c36dec1c084253989eb5196be51b</a>
- ETCCDI/CRD Climate Change Indices (2013).
   http://etccdi.pacificclimate.org/index.shtml
- Meteogalicia. <a href="https://www.meteogalicia.gal/web/index.action">https://www.meteogalicia.gal/web/index.action</a>
- Schramm, M. (2018). Mann-Kendall Trend. GitHub.
   https://github.com/mps9506/Mann-Kendall-Trend/blob/master/mk\_test.py

## **ANEXO A**

# A.1 Cálculo anual de los Índices de temperatura extrema

import pandas as pd import numpy as np import matplotlib.pyplot as plt import datetime as dt from scipy import stats

#### # DATOS DE TMAX Y TMIN

tmax =

pd.read\_csv('0\_Tmaximas.csv',sep=";",parse\_dates=[0],dayfirst=True,index\_col=['Fecha']) tmin =

pd.read\_csv('0\_Tminimas.csv',sep=";",parse\_dates=[0],dayfirst=True,index\_col=['Fecha'])

### # TEMPERATURAS MAXIMAS

dias\_anho\_tmax = tmax.groupby((pd.Grouper(freq='Y'))).count() #Recuento de días de cada año

dias\_anho\_tmax.index = dias\_anho\_tmax.index.year #Cambio índice a años

#### # TEMPERATURAS MÍNIMAS

dias\_anho\_tmin = tmin.groupby((pd.Grouper(freq='Y'))).count() #Recuento de días de cada año dias\_anho\_tmin.index = dias\_anho\_tmin.index.year #Cambio índice a años

# INDICE FECHAS (AÑOS)

```
indices = dias_anho_tmax.index
#NOMBRE ESTACIONES
nb = tmax.columns
estaciones = ['Allariz', 'As Pontes', 'Coruña', 'Coruña aeropuerto', 'Lavacolla', 'Lourizan', 'Lugo',
'Ourense', 'Paramos-Guillarei', 'Vigo', 'Xinzo']
# CREA TRABLAS DONDE ALMACENAR LOS RESULTADOS
indx = ['TX90','TX10','TN90','TN10','TXx','TXn','TNx','TNn','FD','ID','SU','TR','WSDI','CSDI','GSL']
for i in indx:
  exec('{}= pd.DataFrame(index=indices,columns=nb)'.format(i))
# INDICES TX90, TX10, TN90, TN10
for i in range (0,len(nb)):
  y=1965 #AÑO INICIAL
  for j in range (0,len(indices)):
     condicion1=((dias_anho_tmax[nb[i]][dias_anho_tmax.index==y]) >= 300).bool()
     if condicion1==True: # EXCLUYE AÑOS CON MENOS DE 300 DATOS
       a = tmax[nb[i]].quantile(0.1,)
       TX10[nb[i]][TX10.index==y] = tmax[nb[i]][tmax.index.year==y][tmax[nb[i]]<=a].count()
       b = tmax[nb[i]].quantile(0.9,)
       TX90[nb[i]][TX90.index==y] = tmax[nb[i]][tmax.index.year==y][tmax[nb[i]]>=b].count()
     condicion2=((dias_anho_tmin[nb[i]][dias_anho_tmin.index==y]) >= 300).bool()
     if condicion2==True: #Solo se calcula el indice si se tienen mas de 300 datos por año
       c = tmin[nb[i]].quantile(0.1,)
       TN10[nb[i]][TN10.index==y] = tmin[nb[i]][tmin.index.year==y][tmin[nb[i]] <= c].count()
       d = tmin[nb[i]].quantile(0.9,)
       TN90[nb[i]][TN90.index==y] = tmin[nb[i]][tmin.index.year==y][tmin[nb[i]]>=d].count()
     y+=1
# INDICES TXx, TXn, TNx, TNn
for i in range (0,len(nb)):
  y = 1965
  for j in range (0,len(indices)):
     condicion1=((dias_anho_tmax[nb[i]][dias_anho_tmax.index==y]) >= 300).bool()
     if condicion1==True:
       e = tmax[nb[i]][tmax[nb[i]].index.year==y].max()
       TXx[nb[i]][TXx[nb[i]].index==y] = e
       f = tmax[nb[i]][tmax[nb[i]].index.year==y].min()
       TXn[nb[i]][TXn[nb[i]].index==y] = f
```

```
condicion2=((dias_anho_tmin[nb[i]][dias_anho_tmin.index==y]) >= 300).bool()
     if condicion2==True:
       g = tmin[nb[i]][tmin[nb[i]].index.year==y].max()
       TNx[nb[i]][TXx[nb[i]].index==y]=g
       h = tmin[nb[i]][tmin[nb[i]].index.year==y].min()
       TNn[nb[i]][TNn[nb[i]].index==y] = h
     y+=1
# INDICES FD, ID, SU, TR
for i in range (0,len(nb)):
  y = 1965
  for j in range (0,len(indices)):
     condicion2=((dias_anho_tmin[nb[i]][dias_anho_tmin.index==y]) >= 300).bool()
     if condicion2==True:
       k = tmin[tmin <= 0][nb[i]][tmin[tmin <= 0][nb[i]].index.year == y].count()
       FD[nb[i]][FD[nb[i]].index==y] = k
       o = tmin[tmin>=20][nb[i]][tmin[tmin>=20][nb[i]].index.year==y].count()
       TR[nb[i]][TR[nb[i]].index==y] = o
     condicion1=((dias_anho_tmax[nb[i]][dias_anho_tmax.index==y]) >= 300).bool()
     if condicion1==True:
       m = tmax[tmax<=0][nb[i]][tmax[tmax<=0][nb[i]].index.year==y].count()
       ID[nb[i]][ID[nb[i]].index==y] = m
       n = tmax[tmax>=25][nb[i]][tmax[tmax>=25][nb[i]].index.year==y].count()
       SU[nb[i]][SU[nb[i]].index==y] = n
     y+=1
# INDICE CSDI
for k in range (0,len(nb)):
  percentil10 = tmin[nb[k]].quantile(0.1,) # Percentil 10
  y = 1965
  for i in range (0,len(indices)):
       condicion2=((dias_anho_tmin[nb[k]][dias_anho_tmin.index==y]) >= 300).bool()
       if condicion2==True:
          I0 = tmin[nb[k]][tmin.index.year==y][tmin[nb[k]]<=percentil10].index # Datos de T >
p10
          m = len(10) # longitud de la lista
          I1 = I0[0:m-1] # nueva lista datos del primero a penultimo
          I2 = I0[1:m] # nueva lista datos del segundo al ultimo
          dl = (I2 - I1).days # diferencia
          long = len(dl) # longitud de la lista de diferencias
```

```
cont = 0 # contador de dias frios
          j=0 # Posicion en la lista dl
          nCSDI = 0 # numero de periodos frios
          while j<(len(dl)-1):
            if dl[j]==1:
               cont+=1 # si diferencia = 1, suma 1 dia frio.
               if cont>=6 and dl[j+1]!=1: # Si 6 dias o mas frios consecutivos y el siguiente no
es frio
                 nCSDI+=1 #Cuenta un periodo CSDI
                 cont=0 #restaura a 0 el contador de días frios para identificar nuevos periodos
frios
            else:
               cont=0
            j+=1
          CSDI[nb[k]][CSDI.index==y] = nCSDI
          y+=1
       else:
          CSDI[nb[k]][CSDI.index==y] = np.nan
          y+=1
# INDICE WSDI
for k in range (0,len(nb)):
  y = 1965
  percentil90 = tmax[nb[k]].quantile(0.9,) # Percentil 90
  for i in range (0,len(indices)):
     condicion1=((dias_anho_tmax[nb[k]][dias_anho_tmax.index==y]) >= 300).bool()
     if condicion1==True:
       10 = tmax[nb[k]][tmax.index.year==y][tmax[nb[k]]>=percentil90].index
       m = len(10)
       11 = 10[0:m-1]
       12 = 10[1:m]
       dI = (I2 - I1).days
       long = len(dl)
       cont = 0
       i=0
       nWSDI = 0
       while j<len(dl)-1:
          if dl[j]==1:
            cont+=1
            if cont>=6 and dl[j+1]!=1:
```

```
nWSDI+=1
              cont=0
         else:
           cont=0
         j+=1
       WSDI[nb[k]][WSDI.index==y] = nWSDI
       y+=1
    else:
       WSDI[nb[k]][WSDI.index==y] = np.nan
       y+=1
# INDICES DTR, ETR
dtr= tmax - tmin
DTR = dtr.groupby(pd.Grouper(freq='Y')).mean()
DTR.index = DTR.index.year
ETR = TXx-TNn
# TEMPERATURA MEDIA PARA CALCULAR EL GSL
tmed = (tmax + tmin)/2
# FUNCIONES PARA EL CALCULO DEL GSL
def primero(lista): #Identifica primer dia de un periodo de 5 dias consecutivos
  I = len(lista)
  fecha=np.nan
  if I>5: #Deben ser 5 días consecutivos
    j = 0 #Recorrer lista
    dias_consecutivos = 0
    while (j+1)!=I:
       dia1 = lista[j]
       dia2 = lista[j+1]
       diferencia = (dia2 - dia1).days
       if diferencia==1:
         dias_consecutivos+=1
         if dias_consecutivos==5: #Condicion para poder guardar el dia
           fecha = dia1-dt.timedelta(days=4)
       else:
         dias_consecutivos = 0
      j+=1
  return fecha
```

```
def calc_GSL(fst,lst): # Calcula duracion del periodo entre dos fechas
  GSL = (Ist-fst).days
  return GSL
# INDICE GSL
for i in range(0,len(nb)):
  y = 1965
  while y<=2014:
     condicion=((dias_anho_tmax[nb[i]][dias_anho_tmax.index==y]) >= 350).bool()
     if condicion==True:
       #Días de T>5°C.
       lista1 = tmed[nb[i]][(tmed.index.year==y) & (tmed.index.month<=7)][tmed[nb[i]]>5].index
       #Días de T<=5°C.
       lista2 = tmed[nb[i]][(tmed.index.year==y) & (tmed.index.month>7)][tmed[nb[i]]<=5].index
       fst = primero(lista1) #Primer dia periodo de 5 dias consecutivos de T>5°C
       Ist = primero(lista2) #Primer dia periodo de 5 dias consecutivos de T<=5°C
       if (pd.isnull(lst)==True):
          lista3 = tmed[nb[i]][(tmed.index.year==y) &
(tmed.index.month<=7)][tmed[nb[i]]<=5].index
          longitud=len(lista3)
          anho = tmed[nb[i]][(tmed.index.year==y)].index
          if longitud>5:
            lst = primero(lista3)
            if (pd.isnull(lst)==True):
               GSL[nb[i]][GSL.index==y] = 365
            else:
               GSL[nb[i]][GSL.index==y] = 365 - (fst-lst).days
          else:
             GSL[nb[i]][GSL.index==y] = 365
       else:
          GSL[nb[i]][GSL.index==y] = calc_GSL(fst,lst)
     y+=1
# Tablas para guardar datos de las regresiones lineales
parametros = ['Slope','r value', 'p value W', 'std error']
indices = ['TX90', 'TX10', 'TN90', 'TN10', 'TXx', 'TXn', 'TNx', 'TNn', 'FD', 'ID', 'SU', 'TR', 'WSDI',
'CSDI', 'GSL', 'DTR', 'ETR']
listatablas= []
for i in indices:
  listatablas.append('lr'+str(i))
for i in listatablas:
```

```
exec('{}= pd.DataFrame(index=nb,columns=parametros)'.format(i))
def regresionlineal(u): # FUNCION REGRESION LINEAL
  u = u[\sim u.isnull()]
  x= u.index
  y= u.tolist()
  slope, intercept, r_value, p_value, std_err = stats.linregress(x,y)
  plt.plot(x, intercept + slope*x, 'k')
  return slope,r_value,p_value,std_err
# INDICE TX90
for i in range (0,len(nb)):
  fig=plt.figure(figsize=[10,5])
  f = TX90[dias_anho_tmax>=300][nb[i]].plot(style='.-',color='r',fontsize=14)
  plt.title('TX90 - '+estaciones[i],fontsize= 15)
  plt.ylabel('Días',fontsize=14)
  plt.xlabel('t(años)',fontsize=14)
# REGRESION LINEAL TX90
  u= TX90[dias anho tmax>=300][nb[i]]
  resultado = regresionlineal(u)
  for n in range (0,len(parametros)):
     IrTX90[parametros[n]].loc[nb[i]]= resultado[n]
  plt.savefig('F_TX90_'+estaciones[i]+'.png')
# INDICE TX10
for i in range (0,len(nb)):
  fig=plt.figure(figsize=[10,5])
  f = TX10[dias_anho_tmax>=300][nb[i]].plot(style='.-',color='r')
  plt.title('TX10 - '+estaciones[i],fontsize= 15)
  plt.ylabel('Días',fontsize=14)
  plt.xlabel('t(años)',fontsize=14)
# REGRESION LINEAL TX10
  u= TX10[dias_anho_tmax>=300][nb[i]]
  resultado = regresionlineal(u)
```

for n in range (0,len(parametros)):

IrTX10[parametros[n]].loc[nb[i]]= resultado[n]

plt.savefig('F\_TX10\_'+estaciones[i]+'.png')

```
# INDICE TN90
for i in range (0,len(nb)):
       fig=plt.figure(figsize=[10,5])
       f = TN90[dias_anho_tmin>=300][nb[i]].plot(style='.-',title='TN90 - '+estaciones[i],color='b')
      f.set_ylabel('Días')
      f.set_xlabel(' ')
# REGRESION LINEAL TN90
       u= TN90[dias_anho_tmin>=300][nb[i]]
       resultado = regresionlineal(u)
       for n in range (0,len(parametros)):
             IrTN90[parametros[n]].loc[nb[i]]= resultado[n]
     plt.savefig('F_TN90_'+estaciones[i]+'.png')
# INDICE TN10
for i in range (0,len(nb)):
       fig=plt.figure(figsize=[10,5])
      f = TN10[dias_anho_tmin>=300][nb[i]].plot(style='.-',title='TN10 - '+estaciones[i],color='b')
      f.set_ylabel('Días')
      f.set_xlabel(' ')
# REGRESION LINEAL TN10
       u= TN10[dias_anho_tmin>=300][nb[i]]
       resultado = regresionlineal(u)
       for n in range (0,len(parametros)):
             IrTN10[parametros[n]].loc[nb[i]]= resultado[n]
       plt.savefig('F_TN10_'+estaciones[i]+'.png')
# INDICE TXx
for i in range(0,len(nb)):
       fig=plt.figure()
      f = TXx[dias\_anho\_tmax>=300][nb[i]].plot(style='.-',color='r',figsize=[12,5],title='TXx-loops-formation for the state of the state of
'+estaciones[i])
      f.set_ylabel('Temperatura (°C)')
      f.set_xlabel(' ')
# REGRESION LINEAL TXx
       u= TXx[dias_anho_tmax>=300][nb[i]]
```

```
resultado = regresionlineal(u)
  for n in range (0,len(parametros)):
     IrTXx[parametros[n]].loc[nb[i]]= resultado[n]
  plt.savefig('F_TXx_'+estaciones[i]+'.png')
# INDICE TXn
for i in range(0,len(nb)):
  fig=plt.figure()
  f = TXn[dias_anho_tmax>=300][nb[i]].plot(style='.-',color='r',figsize=[12,5],title='TXn -
'+estaciones[i])
  f.set_ylabel('Temperatura (°C)')
  f.set_xlabel(' ')
# REGRESION LINEAL TXn
  u= TXn[dias_anho_tmax>=300][nb[i]]
  resultado = regresionlineal(u)
  for n in range (0,len(parametros)):
     IrTXn[parametros[n]].loc[nb[i]]= resultado[n]
  plt.savefig('F_TXn_'+estaciones[i]+'.png')
# INDICE TNx
for i in range(0,len(nb)):
  fig=plt.figure()
  f = TNx[dias_anho_tmin>=300][nb[i]].plot(style='.-',color='b',figsize=[12,5],title='TNx -
'+estaciones[i])
  f.set_ylabel('Temperatura (°C)')
  f.set_xlabel(' ')
# REGRESION LINEAL TNx
  u= TNx[dias_anho_tmin>=300][nb[i]]
  resultado = regresionlineal(u)
  for n in range (0,len(parametros)):
     IrTNx[parametros[n]].loc[nb[i]]= resultado[n]
  plt.savefig('F_TNx_'+estaciones[i]+'.png')
# INDICE TNn
for i in range(0,len(nb)):
  fig=plt.figure()
  f = TNn[dias_anho_tmin>=300][nb[i]].plot(style='.-',color='b',figsize=[12,5],title='TNn -
'+estaciones[i])
```

```
f.set_ylabel('Temperatura (°C)')
  f.set xlabel('')
# REGRESION LINEAL TNn
  u= TNn[dias anho tmin>=300][nb[i]]
  resultado = regresionlineal(u)
  for n in range (0,len(parametros)):
     IrTNn[parametros[n]].loc[nb[i]]= resultado[n]
  plt.savefig('F_TNn_'+estaciones[i]+'.png')
# INDICE FD
for i in range(0,len(nb)):
  fig=plt.figure()
  f = FD[dias_anho_tmin>=300][nb[i]].plot(style='.-',color='b',figsize=[12,5],title='FD -
'+estaciones[i])
  f.set_ylabel('Días')
  f.set_xlabel(' ')
# REGRESION LINEAL FD
  u= FD[dias_anho_tmin>=300][nb[i]]
  resultado = regresionlineal(u)
  for n in range (0,len(parametros)):
     IrFD[parametros[n]].loc[nb[i]]= resultado[n]
  plt.savefig('F_FD_'+estaciones[i]+'.png')
# INDICE ID
for i in range(0,len(nb)):
  fig=plt.figure()
  f = ID[dias_anho_tmax>=300][nb[i]].plot(style='.-',color='b',figsize=[12,2],title='ID -
'+estaciones[i])
  f.set_ylabel('Días')
  f.set_xlabel(' ')
# REGRESION LINEAL ID
  u= ID[dias_anho_tmax>=300][nb[i]]
  resultado = regresionlineal(u)
  for n in range (0,len(parametros)):
     IrID[parametros[n]].loc[nb[i]]= resultado[n]
  plt.savefig('F_ID_'+estaciones[i]+'.png')
```

# INDICE SU

```
for i in range(0,len(nb)):
  fig=plt.figure()
  f = SU[dias_anho_tmax>=300][nb[i]].plot(style='.-',color='r',figsize=[12,5],title='SU -
'+estaciones[i])
  f.set_ylabel('Días')
  f.set_xlabel(' ')
# REGRESION LINEAL SU
  u= SU[dias_anho_tmax>=300][nb[i]]
  resultado = regresionlineal(u)
  for n in range (0,len(parametros)):
     IrSU[parametros[n]].loc[nb[i]]= resultado[n]
  plt.savefig('F_SU_'+estaciones[i]+'.png')
# INDICE TR
for i in range(0,len(nb)):
  fig=plt.figure()
  f = TR[dias_anho_tmin>=300][nb[i]].plot(style='.-',color='b',figsize=[10,5],title='TR -
'+estaciones[i])
  f.set_ylabel('Días')
  f.set_xlabel(' ')
# REGRESION LINEAL TR
  u= TR[dias_anho_tmin>=300][nb[i]]
  resultado = regresionlineal(u)
  for n in range (0,len(parametros)):
     IrTR[parametros[n]].loc[nb[i]]= resultado[n]
  plt.savefig('F_TR_'+estaciones[i]+'.png')
# INDICE WSDI
for i in range(0,len(nb)):
  fig=plt.figure()
  f = WSDI[dias_anho_tmax>=300][nb[i]].plot(style='.-',color='r',figsize=[12,5],title='WSDI -
'+estaciones[i])
  f.set ylabel('Periodo (6 días)')
  f.set_xlabel(' ')
  f.set_ylim(0,15)
# REGRESION LINEAL WSDI
  u= WSDI[dias_anho_tmin>=300][nb[i]]
```

```
resultado = regresionlineal(u)
  for n in range (0,len(parametros)):
     IrWSDI[parametros[n]].loc[nb[i]]= resultado[n]
  plt.savefig('F_WSDI_'+estaciones[i]+'.png')
# INDICE CSDI
for i in range(0,len(nb)):
  fig=plt.figure()
  f = CSDI[dias_anho_tmin>=300][nb[i]].plot(style='.-',color='b',figsize=[12,2],title='CSDI -
'+estaciones[i])
  f.set_ylabel('Periodo (6 días)')
  f.set_xlabel(' ')
# REGRESION LINEAL CSDI
  u= CSDI[dias_anho_tmin>=300][nb[i]]
  resultado = regresionlineal(u)
  for n in range (0,len(parametros)):
     lrCSDI[parametros[n]].loc[nb[i]]= resultado[n]
  plt.savefig('F_CSDI_'+estaciones[i]+'.png')
# INDICE GSL
for i in range(0,len(nb)):
  fig=plt.figure()
  f = GSL[nb[i]].plot(style='.-',color='b',figsize=[12,6],title='GSL - '+estaciones[i])
  f.set_ylabel('Periodo (días)')
  f.set_xlabel(' ')
# REGRESION LINEAL GSL
  u= GSL[nb[i]]
  resultado = regresionlineal(u)
  for n in range (0,len(parametros)):
     lrGSL[parametros[n]].loc[nb[i]]= resultado[n]
  plt.savefig('F_GSL_'+estaciones[i]+'.png')
# INDICE DTR
for i in range(0,len(nb)):
  fig=plt.figure()
  f = DTR[dias_anho_tmax>=300][nb[i]].plot(style='.-',color='g',figsize=[12,5],title='DTR -
'+estaciones[i])
  f.set_ylabel('Tmperatura (°C)')
  f.set_xlabel(' ')
```

```
# REGRESION LINEAL DTR
  u= DTR[dias_anho_tmax>=300][nb[i]]
  resultado = regresionlineal(u)
  for n in range (0,len(parametros)):
     IrDTR[parametros[n]].loc[nb[i]]= resultado[n]
  plt.savefig('F_DTR_'+estaciones[i]+'.png')
# INDICE ETR
for i in range(0,len(nb)):
  fig=plt.figure()
  f = ETR[nb[i]].plot(style='.-',color='g',figsize=[12,5],title='ETR - '+estaciones[i])
  f.set_ylabel('Temperatura (°C)')
  f.set_xlabel(' ')
# REGRESION LINEAL ETR
  u= ETR[dias_anho_tmax>=300][nb[i]]
  resultado = regresionlineal(u)
  for n in range (0,len(parametros)):
     IrETR[parametros[n]].loc[nb[i]]= resultado[n]
  plt.savefig('ETR_'+estaciones[i]+'.png')
# TEST MANN-KENDALL (Schramm, 2018)
from scipy.stats import norm
def mk_test(x, alpha=0.05):
  n = len(x)
  s = 0
  for k in range(n-1):
    for j in range(k+1, n):
       s += np.sign(x[j] - x[k]) # Calcula S
  unique_x = np.unique(x) # Calcula valores unicos
  g = len(unique_x)
  if n == g:
    var s = (n*(n-1)*(2*n+5))/18 \# calcula VAR(S) si no hay valores repetidos
  else:
    tp = np.zeros(unique_x.shape)
    for i in range(len(unique_x)):
       tp[i] = sum(x == unique_x[i]) #calcula el nº de veces que se repite cada valor "unico"
```

```
var_s = (n^*(n-1)^*(2^*n+5) - np.sum(tp^*(tp-1)^*(2^*tp+5)))/18 # si hay valores repetidos en la
serie
   if s > 0:
     z = (s - 1)/np.sqrt(var_s)
  elif s < 0:
     z = (s + 1)/np.sqrt(var_s)
  else: \# s == 0:
     z = 0
  p = 2*(1-norm.cdf(abs(z))) # calcula el p value
  h = abs(z) > norm.ppf(1-alpha/2)
  if (z < 0) and h:
     trend = 'Decreasing'
  elif (z > 0) and h:
     trend = 'Increasing'
  else:
     trend = 'no trend'
  return trend, p
# Funcion que aplica el Test de MannKendall a una serie de datos y devuelve el resultado
def trends(df,parametro):
  columna = []
  for i in range (0,len(nb)): # Crea 1 columna de nan con el mismo numero de filas que de
estaciones
     columna.append(np.nan)
  df['p value MK'] = columna #crea una columna nueva
  df['Trend MK'] = columna #crea una columna nueva
  for i in range (0,len(nb)):
     datos = parametro[nb[i]][pd.notna(parametro[nb[i]])] #datos a los que se les aplica MK
     x = datos.tolist() #Crea lista de valores
     trend, p = mk_t(x,0.05)
     df['p value MK'][nb[i]] = p
     df['Trend MK'][nb[i]] = trend
  return df
# Pasa los datos a la función que aplica el test y guarda los resultados en las tablas indicadas.
tablas = [IrTX90, IrTX10, IrTN90, IrTN10, IrTXx, IrTXn, IrTNx, IrTNn, IrFD, IrID, IrID, IrSU, IrTR,
IrWSDI, IrCSDI, IrGSL, IrDTR, IrETR]
indices = [TX90, TX10, TN90, TN10, TXx, TXn, TNx, TNn, FD, ID, ID, SU, TR, WSDI, CSDI, GSL,
DTR, ETRI
for i in range (0,len(tablas)):
```

```
a = tablas[i]
  b = indices[i]
  trends(a,b)
#Guardar datos
IrTX90.to_csv('Tendencia_TX90.csv',sep=";")
IrTX10.to csv('Tendencia TX10.csv',sep=";")
IrTN90.to_csv('Tendencia_TN90.csv',sep=";")
IrTN10.to_csv('Tendencia_TN10.csv',sep=";")
IrTXx.to_csv('Tendencia_TXx.csv',sep=";")
IrTXn.to_csv('Tendencia_TXn.csv',sep=";")
IrTNx.to_csv('Tendencia_TNx.csv',sep=";")
IrTNn.to_csv('Tendencia_TNn.csv',sep=";")
IrFD.to_csv('Tendencia_FD.csv',sep=";")
IrID.to_csv('Tendencia_ID.csv',sep=";")
IrSU.to_csv('Tendencia_SU.csv',sep=";")
IrTR.to_csv('Tendencia_TR.csv',sep=";")
IrWSDI.to_csv('Tendencia_WSDI.csv',sep=";")
lrCSDI.to_csv('Tendencia_CSDI.csv',sep=";")
IrGSL.to_csv('Tendencia_GSL.csv',sep=";")
IrDTR.to_csv('Tendencia_DTR.csv',sep=";")
IrETR.to_csv('Tendencia_ETR.csv',sep=";")
```

# A.2 Cálculo estacional de los Índices de temperatura extrema

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from scipy import stats

# DATOS DE TMAX Y TMIN

tmax =
pd.read_csv('0_Tmaximas.csv',sep=";",parse_dates=[0],dayfirst=True,index_col=['Fecha'])

tmin =
pd.read_csv('0_Tminimas.csv',sep=";",parse_dates=[0],dayfirst=True,index_col=['Fecha'])

#Nombre de las estaciones
nombre = tmax.columns
estaciones = ['Allariz','As Pontes','Coruna','Coruna Aeropuerto', 'Lavacolla', 'Lourizan', 'Lugo',
'Ourense', 'Paramos-Guillarei', 'Vigo', 'Xinzo']
```

```
# INDICE FECHAS (AÑOS)
var = tmax.index.year
ind = set(var)
# Crea las talas donde se van a almacenar los valores de los indices calculados
indices =
['TX90','TX10','TN90','TN10','TXx','TXn','TNx','TNn','FD','ID','SU','TR','WSDI','CSDI','DTR']
listatablas= []
for i in indices:
  listatablas.append('P'+str(i))
  listatablas.append('V'+str(i))
  listatablas.append('O'+str(i))
  listatablas.append('I'+str(i))
for i in listatablas:
  exec('{}= pd.DataFrame(index=ind, columns=nombre)'.format(i))
# Calcula nº de datos nulos que hay en una serie de datos
def nulos(trimestre):
  n_nan = trimestre.isnull().sum()
  return n_nan
# INDICES TX90, TX10 - Calculo percentiles
P_max = tmax[(tmax.index.month >= 3) & (tmax.index.month <=5)]
V_{max} = tmax[(tmax.index.month >= 6) & (tmax.index.month <= 8)]
O_max = tmax[(tmax.index.month >= 9) & (tmax.index.month <=11)]
I_max = tmax[( (tmax.index.month==12) | ((tmax.index.month<=2) & (tmax.index.month>=1) &
(tmax.index.year>1965)) )]
p10max = P_max.quantile(0.1,) # Percentil 10 de Tmax
v10max = V_max.quantile(0.1,)
o10max = O_max.quantile(0.1,)
i10max = I_max.quantile(0.1,)
p90max = P_max.quantile(0.9,) # Percentil 90 de Tmax
v90max = V_max.quantile(0.9,)
o90max = O_max.quantile(0.9,)
i90max = I max.quantile(0.9,)
# INDICES TX10, TX90 - PVO
y = 1965 #primer año
f = y+len(ind) #último año
while y<f:
```

```
P = tmax[((tmax.index.month >= 3) & (tmax.index.month <=5))& (tmax.index.year == y)]
  V = tmax[((tmax.index.month >= 6) & (tmax.index.month <= 8)) & (tmax.index.year == y)]
  O = tmax[((tmax.index.month >= 9) & (tmax.index.month <=11))& (tmax.index.year == y)]
  prim90 = P[P>=p90max].count() # Dias de Tmax por debajo del percentil 90
  ver90 = V[V>=v90max].count()
  ot90 = O[O >= o90max].count()
  prim10 = P[P<=p10max].count() # Dias de Tmax por debajo del percentil 10
  ver10 = V[V \le v10max].count()
  ot10 = O[O \le 10max].count()
  n_nanP = nulos(P) #Calcula nº nan
  n nanV = nulos(V)
  n_n = nulos(O)
  n_nanP = (n_nanP<=15) # True/False en cada estacion segun si hay o no mas de 15 NaN
  n_n = (n_n = 15)
  n_n = (n_n = 0)
  for i in nombre:
    if n_nanP[i]!=True: #Si hay más de 15 NaN, no calculo nada
       prim10[i] = np.nan #En las estaciones donde haya más de 15 nan elimina el dato
       prim90[i] = np.nan
    if n_nanV[i]!=True:
       ver10[i] = np.nan
       ver90[i] = np.nan
    if n nanO[i]!=True:
       ot10[i] = np.nan
       ot90[i] = np.nan
  for i in range(0,len(prim10)):
     PTX10[nombre[i]][PTX10.index==y] = prim10[i]
     PTX90[nombre[i]][PTX90.index==y] = prim90[i]
    VTX10[nombre[i]][VTX10.index==y] = ver10[i]
    VTX90[nombre[i]][VTX90.index==y] = ver90[i]
    OTX10[nombre[i]][OTX10.index==y] = ot10[i]
    OTX90[nombre[i]][OTX90.index==y] = ot90[i]
 y+=1
# INDICES TX10, TX90 - I
y = 1966
f = y+len(ind)
while y<f-1:
  I = tmax[( ((tmax.index.month==12) & (tmax.index.year == y-1)) | ((tmax.index.month<=2) &
(tmax.index.month>=1) & (tmax.index.year==y)) )]
```

```
inv90 = I[I>=i90max].count() #Dias por encima del percentil 90
  inv10 = I[I<=i10max].count() #Dias por debajo del percentil 10
  n_nanl = nulos(I) #Calculo número de nan
  n nanl = (n nanl <= 15)
  for i in nombre:
     if n nanl[i]!=True:
       inv10[i] = np.nan # Si nº datos < 15 se elimina el dato
       inv90[i] = np.nan
  for i in range(0,len(prim10)):
     ITX10[nombre[i]][ITX10.index==y] = inv10[i]
     ITX90[nombre[i]][ITX90.index==y] = inv90[i]
  y+=1
# INDICES TN90, TN10 - Calculo percentiles
P_min = tmin[(tmin.index.month >= 3) & (tmin.index.month <=5)]
V_min = tmin[(tmin.index.month >= 6) & (tmin.index.month <=8)]
O_min = tmin[(tmin.index.month >= 9) & (tmin.index.month <=11)]
I_min = tmin[( (tmin.index.month==12) | ((tmin.index.month<=2) & (tmin.index.month>=1) &
(tmin.index.year>1965)) )]
p10min = P_min.quantile(0.1,)
v10min = V_min.quantile(0.1,)
o10min = O_min.quantile(0.1,)
i10min = I_min.quantile(0.1,)
p90min = P_min.quantile(0.9,)
v90min = V_min.quantile(0.9,)
o90min = O_min.quantile(0.9,)
i90min = I_min.quantile(0.9,)
# INDICES TN10, TN90 - PVO
y = 1965 #empiezo en el primer año
f = y+len(ind) #último año
while y<f:
  P = tmin[((tmin.index.month >= 3) & (tmin.index.month <=5))& (tmin.index.year == y)]
  V = tmin[((tmin.index.month >= 6) & (tmin.index.month <=8))& (tmin.index.year == y)]
  O = tmin[((tmin.index.month >= 9) & (tmin.index.month <=11))& (tmin.index.year == y)]
  prim90 = P[P>=p90min].count() #Dias por encima del percentil 90
  ver90 = V[V >= v90min].count()
  ot90 = O[O >= o90min].count()
```

```
prim10 = P[P<=p10min].count() #Dias por debajo del percentil 10
  ver10 = V[V \le v10min].count()
  ot10 = O[O \le 010min].count()
  n nanP = nulos(P) #Calcula nº de nan
  n_nanV = nulos(V)
  n_n = nulos(O)
  n_n = (n_n = 15)
  n_n = (n_n = 15)
  n nanO = (n nanO <= 15)
 for i in nombre:
     if n_nanP[i]!=True:
       prim10[i] = np.nan # Si nº datos < 15 se elimina el dato
       prim90[i] = np.nan
  for i in nombre:
     if n_nanV[i]!=True:
       ver10[i] = np.nan
       ver90[i] = np.nan
  for i in nombre:
     if n_nanO[i]!=True:
       ot10[i] = np.nan
       ot90[i] = np.nan
for i in range(0,len(prim10)):
     PTN10[nombre[i]][PTN10.index==y] = prim10[i]
     PTN90[nombre[i]][PTN90.index==y] = prim90[i]
     VTN10[nombre[i]][VTN10.index==y] = ver10[i]
    VTN90[nombre[i]][VTN90.index==y] = ver90[i]
    OTN10[nombre[i]][OTN10.index==y] = ot10[i]
     OTN90[nombre[i]][OTN90.index==y] = ot90[i]
  y+=1
# INDICES TN10, TN90 - I
y = 1966
f = y + len(ind)
while y<f-1:
  I = tmin[( ((tmin.index.month==12) & (tmin.index.year == y-1)) | ((tmin.index.month<=2) &
  (tmin.index.month>=1) & (tmin.index.year==y)) )]
  inv90 = I[I>=i90min].count() #Dias por encima del percentil 90
  inv10 = I[I<=i10min].count() #Dias por debajo del percentil 10
  n_nanl = nulos(I) #Calcula no nan
  n_n = (n_n = 15)
```

```
for i in nombre:
     if n nanl[i]!=True:
       inv10[i] = np.nan # Si nº datos < 15 se elimina el dato
       inv90[i] = np.nan
  for i in range(0,len(prim10)):
     ITN10[nombre[i]][ITN10.index==y] = inv10[i]
     ITN90[nombre[i]][ITN90.index==y] = inv90[i]
  y+=1
# INDICES TXx, TXn, TNx, TNn
def maxima(trimestre):
  n nan = nulos(trimestre)
  if n_nan<=15:
     maximo = trimestre.max()
  else:
     maximo = np.nan
  return máximo
def minima(trimestre):
  n_nan = nulos(trimestre)
  if n_nan<=15:
     minimo = trimestre.min()
  else:
     minimo = np.nan
  return minimo
# INDICES TXx, TXn, TNx, TNn - PVO
y = 1965
f = y + len(ind)
while y<f:
  Pn = tmin[((tmin.index.month >= 3) & (tmin.index.month <=5))& (tmin.index.year == y)]
  Vn = tmin[((tmin.index.month >= 6) & (tmin.index.month <=8))& (tmin.index.year == y)]
  On = tmin[((tmin.index.month >= 9) & (tmin.index.month <=11))& (tmin.index.year == y)]
  Px = tmax[((tmax.index.month >= 3) & (tmax.index.month <= 5)) & (tmax.index.year == y)]
  Vx = tmax[((tmax.index.month >= 6) & (tmax.index.month <= 8)) & (tmax.index.year == y)]
  Ox = tmax[((tmax.index.month >= 9) & (tmax.index.month <=11)) & (tmax.index.year == y)]
  PvaloresTXx = Px.apply(maxima)
  VvaloresTXx = Vx.apply(maxima)
  OvaloresTXx = Ox.apply(maxima)
  PvaloresTXn = Px.apply(minima)
  VvaloresTXn = Vx.apply(minima)
  OvaloresTXn = Ox.apply(minima)
```

```
PvaloresTNx = Pn.apply(maxima)
  VvaloresTNx = Vn.apply(maxima)
  OvaloresTNx = On.apply(maxima)
  PvaloresTNn = Pn.apply(minima)
  VvaloresTNn = Vn.apply(minima)
  OvaloresTNn = On.apply(minima)
  for i in range(0,len(prim10)):
     PTXx[nombre[i]][PTXx.index==y] = PvaloresTXx[i]
    VTXx[nombre[i]][VTXx.index==y] = VvaloresTXx[i]
    OTXx[nombre[i]][OTXx.index==y] = OvaloresTXx[i]
     PTXn[nombre[i]][PTXn.index==y] = PvaloresTXn[i]
    VTXn[nombre[i]][VTXn.index==y] = VvaloresTXn[i]
    OTXn[nombre[i]][OTXn.index==y] = OvaloresTXn[i]
     PTNx[nombre[i]][PTNx.index==y] = PvaloresTNx[i]
    VTNx[nombre[i]][VTNx.index==y] = VvaloresTNx[i]
    OTNx[nombre[i]][OTNx.index==y] = OvaloresTNx[i]
     PTNn[nombre[i]][PTNn.index==y] = PvaloresTNn[i]
    VTNn[nombre[i]][VTNn.index==y] = VvaloresTNn[i]
    OTNn[nombre[i]][OTNn.index==y] = OvaloresTNn[i]
  y+=1
#TXx, TXn, TNx, TNn - I
y = 1966
f = y + len(ind)
while y<f-1:
  In = tmin[( ((tmin.index.month==12) & (tmin.index.year == y-1)) | ((tmin.index.month<=2) &
  (tmin.index.month>=1) & (tmin.index.year==y)) )]
  lx = tmax[( ((tmax.index.month==12) & (tmax.index.year == y-1)) | ((tmax.index.month<=2) &</pre>
  (tmax.index.month>=1) & (tmax.index.year==y)) )]
  IvaloresTXx = Ix.apply(maxima)
  IvaloresTXn = Ix.apply(minima)
  IvaloresTNx = In.apply(maxima)
  IvaloresTNn = In.apply(minima)
  for i in range(0,len(prim10)):
    ITXx[nombre[i]][ITXx.index==y] = IvaloresTXx[i]
    ITXn[nombre[i]][ITXn.index==y] = IvaloresTXn[i]
    ITNx[nombre[i]][ITNx.index==y] = IvaloresTNx[i]
    ITNn[nombre[i]][ITNn.index==y] = IvaloresTNn[i]
  y+=1
# INDICES FD, TR, ID, SU
```

```
def FD(trimestre):
  n_nan = nulos(trimestre)
  if n_nan<=15:
     resultado = trimestre[trimestre<=0].count()
  else:
     resultado = np.nan
  return resultado
def TR(trimestre):
  n_nan = nulos(trimestre)
  if n_nan<=15:
     TR = trimestre[trimestre>=20].count()
     TR = np.nan
  return TR
def ID(trimestre):
  n_nan = nulos(trimestre)
  if n_nan<=15:
     ID = trimestre[trimestre<=0].count()</pre>
  else:
     ID = np.nan
  return ID
def SU(trimestre):
  n_nan = nulos(trimestre)
  if n_nan<=15:
     SU = trimestre[trimestre>=25].count()
  else:
     SU = np.nan
  return SU
# INDICES F, TR, ID, SU - PVO
y = 1965
f = y + len(ind)
while y<f:
  Pn = tmin[((tmin.index.month >= 3) & (tmin.index.month <=5))& (tmin.index.year == y)]
  Vn = tmin[((tmin.index.month >= 6) & (tmin.index.month <=8))& (tmin.index.year == y)]
  On = tmin[((tmin.index.month >= 9) & (tmin.index.month <=11))& (tmin.index.year == y)]
  Px = tmax[((tmax.index.month >= 3) & (tmax.index.month <= 5)) & (tmax.index.year == y)]
```

```
Vx = tmax[((tmax.index.month >= 6) & (tmax.index.month <= 8)) & (tmax.index.year == y)]
  Ox = tmax[((tmax.index.month >= 9) & (tmax.index.month <=11)) & (tmax.index.year == y)]
  valorPFD = Pn.apply(FD) #Calculo Índice FD
  valorVFD = Vn.apply(FD)
  valorOFD = On.apply(FD)
  valorPTR = Pn.apply(TR) # Calculo ínidce TR
  valorVTR = Vn.apply(TR)
  valorOTR = On.apply(TR)
  valorPID = Px.apply(ID) # Calculo índice ID
  valorVID = Vx.apply(ID)
  valorOID = Ox.apply(ID)
  valorPSU = Px.apply(SU) # Calculo índice SU
  valorVSU = Vx.apply(SU)
  valorOSU = Ox.apply(SU)
  for i in range(0,len(valorPFD)):
     PFD[nombre[i]][PFD.index==y] = valorPFD[i]
     VFD[nombre[i]][VFD.index==y] = valorVFD[i]
     OFD[nombre[i]][OFD.index==y] = valorOFD[i]
     PTR[nombre[i]][PTR.index==y] = valorPTR[i]
     VTR[nombre[i]][VTR.index==y] = valorVTR[i]
     OTR[nombre[i]][OTR.index==y] = valorOTR[i]
     PID[nombre[i]][PID.index==y] = valorPID[i]
     VID[nombre[i]][VID.index==y] = valorVID[i]
     OID[nombre[i]][OID.index==y] = valorOID[i]
     PSU[nombre[i]][PSU.index==y] = valorPSU[i]
     VSU[nombre[i]][VSU.index==y] = valorVSU[i]
     OSU[nombre[i]][OSU.index==y] = valorOSU[i]
  y+=1
# INDICES F, TR, ID, SU - I
y = 1966
f = y + len(ind)
while y<f-1:
  In = tmin[( ((tmin.index.month==12) & (tmin.index.year == y-1)) | ((tmin.index.month<=2) &
  (tmin.index.month>=1) & (tmin.index.year==y)) )]
  Ix = tmax[( ((tmax.index.month==12) & (tmax.index.year == y-1)) | ((tmax.index.month<=2) &
  (tmax.index.month>=1) & (tmax.index.year==y)) )]
  valorIFD = In.apply(FD)
  valorITR = In.apply(TR)
  valorIID = Ix.apply(ID)
```

```
valorISU = Ix.apply(SU)
  for i in range(0,len(prim10)):
     IFD[nombre[i]][IFD.index==y] = valorIFD[i]
     ITR[nombre[i]][ITR.index==y] = valorITR[i]
     IID[nombre[i]][IID.index==y] = valorIID[i]
     ISU[nombre[i]][ISU.index==y] = valorISU[i]
  y+=1
# INDICE WSDI
def listadif(lista):
  I = len(lista)
  lista1 = lista[0:l-1]
  lista2 = lista[1:l]
  dif = (lista2-lista1).days #dias de diferencia entre una fecha y la siguiente
  long = len(dif) # calculo longitud de la lista de dias
  return dif, long #me devuelve la lista de dias y su longitud
def calculaWSDI(listadatos):
  dif, I = listadif(listadatos) #usa la lista de fechas
  cont = 0
  WSDI = 0 #Numero de periodos
  j=0
  while j<(I-1):
     if dif[j]==1:
       cont+=1
       if cont>=6 & dif[j+1]!=1:
          WSDI = cont
          cont=0
     j+=1
  return WSDI
# INDICE WSDI - PVO
y = 1965
f = y + len(ind)
while y<=f:
  for i in range(0,len(nombre)):
     Px = tmax[((tmax.index.month >= 3) & (tmax.index.month <= 5)) & (tmax.index.year == y)]
     Vx = tmax[((tmax.index.month >= 6) & (tmax.index.month <= 8)) & (tmax.index.year == y)]
     Ox = tmax[((tmax.index.month >= 9) & (tmax.index.month <= 11)) & (tmax.index.year == y)]
     n_n = nulos(Px)
```

```
n_nanV = nulos(Vx)
     n_n = nulos(Ox)
     if n_nanP[i]<=15:
       datosP= (Px[nombre[i]][Px[nombre[i]]>p90max[i]])
       fechas = datosP.index
       pWSDI = calculaWSDI(fechas)
     else:
       pWSDI = np.nan
     if n_nanV[i]<=15:
       datosV= (Vx[nombre[i]][Vx[nombre[i]]>v90max[i]])
       fechas = datosV.index
       vWSDI = calculaWSDI(fechas)
     else:
       vWSDI = np.nan
     if n_nanP[i]<=15:
       datosO= (Ox[nombre[i]][Ox[nombre[i]]>o90max[i]])
       fechas = datosO.index
       oWSDI = calculaWSDI(fechas)
     else:
       oWSDI = np.nan
     PWSDI[nombre[i]][PWSDI.index == y] = pWSDI
    VWSDI[nombre[i]][VWSDI.index == y] = vWSDI
    OWSDI[nombre[i]][OWSDI.index == y] = oWSDI
  y+=1
# INDICE WSDI - I
y = 1966
f = y + len(ind)
while y<f-1:
  for i in range(0,len(nombre)):
     lx = tmax[( ((tmax.index.month==12) & (tmax.index.year == y-1)) | ((tmax.index.month<=2)</pre>
& (tmax.index.month>=1) & (tmax.index.year==y)) )]
     n_nanl = nulos(lx)
     if n_nanl[i]<=15:
       datos = (lx[nombre[i]][lx[nombre[i]]>i90max[i]])
       fechas = datos.index
       iWSDI = calculaWSDI(fechas)
     else:
       iWSDI = np.nan
     IWSDI[nombre[i]][IWSDI.index == y] = iWSDI
```

```
y+=1
# INDICE CSDI
def calculaCSDI(listadatos):
  dif, I = listadif(listadatos) # recibe la lista de fechas
  cont = 0
  CSDI = 0 #Numero de periodos
  j=0
  while j<(l-1):
     if dif[j]==1:
       cont+=1
       if cont>=6 & dif[j+1]!=1:
          CSDI = cont
          cont=0
    j+=1
  return CSDI
# INDICE CDSI - PVO
y = 1965
f = y + len(ind)
while y<f-1:
  for i in range(0,len(nombre)):
     Pn = tmin[((tmin.index.month >= 3) & (tmin.index.month <=5))& (tmin.index.year == y)]
     Vn = tmin[((tmin.index.month >= 6) & (tmin.index.month <=8))& (tmin.index.year == y)]
     On = tmin[((tmin.index.month >= 9) & (tmin.index.month <=11))& (tmin.index.year == y)]
     n_nanP = nulos(Pn)
     n_nanV = nulos(Vn)
     n_nanO = nulos(On)
    if n_nanP[i]<=15:
       datosp = (Pn[nombre[i]][Pn[nombre[i]]<p10min[i]])
       fechas = datosp.index
       CSDIp = calculaCSDI(fechas)
     else:
       CSDIp = np.nan
     if n nanV[i]<=15:
       datosv = (Vn[nombre[i]][Vn[nombre[i]]<v10min[i]])
       fechas = datosv.index
       CSDIv = calculaCSDI(fechas)
```

else:

CSDIv = np.nan

```
if n_nanO[i]<=15:
       datoso = (On[nombre[i]][On[nombre[i]]<o10min[i]])
       fechas = datoso.index
       CSDIo = calculaCSDI(fechas)
     else:
       CSDIo = np.nan
     PCSDI[nombre[i]][PCSDI.index == y] = CSDIp
    VCSDI[nombre[i]][VCSDI.index == y] = CSDIv
    OCSDI[nombre[i]][OCSDI.index == y] = CSDIo
  y+=1
# INDICE CSDI
y = 1966
f = y + len(ind)
while y<f-1:
  for i in range(0,len(nombre)):
     In = tmin[( ((tmin.index.month==12) & (tmin.index.year == y-1)) | ((tmin.index.month<=2) &
(tmin.index.month>=1) & (tmin.index.year==y)) )]
     n_nanl = nulos(ln)
     if n_nanl[i]<=15:
       datos = In[nombre[i]][In[nombre[i]]<i10min[i]]
       fechas = datos.index
       iCSDI = calculaCSDI(fechas)
     else:
       iCSDI = np.nan
     ICSDI[nombre[i]][ICSDI.index == y] = iCSDI
  y+=1
# INDICE ETR
PETR = PTXx - PTNn
VETR = VTXx - VTNn
OETR = OTXx - OTNx
IETR = ITXx - ITNn
# INDICE DTR
v = 1965
f = y + len(ind)
while y<f:
  Pn = tmin[((tmin.index.month >= 3) & (tmin.index.month <=5))& (tmin.index.year == y)]
  Vn = tmin[((tmin.index.month >= 6) & (tmin.index.month <=8))& (tmin.index.year == y)]
  On = tmin[((tmin.index.month >= 9) & (tmin.index.month <=11))& (tmin.index.year == y)]
```

```
Px = tmax[((tmax.index.month >= 3) & (tmax.index.month <=5))& (tmax.index.year == y)]
  Vx = tmax[((tmax.index.month >= 6) & (tmax.index.month <= 8)) & (tmax.index.year == y)]
  Ox = tmax[((tmax.index.month >= 9) & (tmax.index.month <=11)) & (tmax.index.year == y)]
  valorPDTR = (Px-Pn).mean()
  valorVDTR = (Vx-Vn).mean()
  valorODTR = (Ox-On).mean()
  for i in range(0,len(valorPDTR)):
     PDTR[nombre[i]][PDTR.index==y] = valorPDTR[i]
     VDTR[nombre[i]][VDTR.index==y] = valorVDTR[i]
     ODTR[nombre[i]][ODTR.index==y] = valorODTR[i]
  y+=1
# INDICE DTR - I
y = 1966
f = y + len(ind)
while y<f-1:
  In = tmin[( ((tmin.index.month==12) & (tmin.index.year == y-1)) | ((tmin.index.month<=2) &
  (tmin.index.month>=1) & (tmin.index.year==y)) )]
  lx = tmax[( ((tmax.index.month==12) & (tmax.index.year == y-1)) | ((tmax.index.month<=2) &</pre>
  (tmax.index.month>=1) & (tmax.index.year==y)) )]
  valorIDTR = (Ix-In).mean()
  for i in range(0,len(valorIDTR)):
     IDTR[nombre[i]][IDTR.index==y] = valorIDTR[i]
  y+=1
# Calculo tendencias
parametros = ['Slope', 'r value', 'p value', 'std error']
indices = ['trendTX90', 'trendTX10', 'trendTN90', 'trendTN10', 'trendTXx', 'trendTXn', 'trendTNx',
'trendTNn', 'trendFD', 'trendID', 'trendSU', 'trendTR', 'trendWSDI', 'trendCSDI', 'trendDTR',
'trendETR']
listatablas= []
for i in indices:
  listatablas.append('P'+str(i))
  listatablas.append('V'+str(i))
  listatablas.append('O'+str(i))
  listatablas.append('I'+str(i))
for i in listatablas:
  exec('{}= pd.DataFrame(index=nombre, columns=parametros)'.format(i))
def regresionlineal(u):
```

```
u = u[\sim u.isnull()]
  x= u.index
  y= u.tolist()
  slope, intercept, r_value, p_value, std_err = stats.linregress(x,y)
  return slope,r_value,p_value,std_err
#TX90
resultado1= PTX90.apply(regresionlineal)
resultado2= VTX90.apply(regresionlineal)
resultado3= OTX90.apply(regresionlineal)
resultado4= ITX90.apply(regresionlineal)
for i in range(0,len(nombre)):
  for j in range(0,len(parametros)):
     PtrendTX90[parametros[j]][i]=resultado1[i][j]
     VtrendTX90[parametros[j]][i]=resultado2[i][j]
     OtrendTX90[parametros[j]][i]=resultado3[i][j]
     ItrendTX90[parametros[j]][i]=resultado4[i][j]
#TX10
resultado1= PTX10.apply(regresionlineal)
resultado2= VTX10.apply(regresionlineal)
resultado3= OTX10.apply(regresionlineal)
resultado4= ITX10.apply(regresionlineal)
for i in range(0,len(nombre)):
  for j in range(0,len(parametros)):
     PtrendTX10[parametros[j]][i]=resultado1[i][j]
     VtrendTX10[parametros[j]][i]=resultado2[i][j]
     OtrendTX10[parametros[j]][i]=resultado3[i][j]
     ItrendTX10[parametros[j]][i]=resultado4[i][j]
#TN90
resultado1= PTN90.apply(regresionlineal)
resultado2= VTN90.apply(regresionlineal)
resultado3 = OTN90.apply(regresionlineal)
resultado4= ITN90.apply(regresionlineal)
for i in range(0,len(nombre)):
  for j in range(0,len(parametros)):
     PtrendTN90[parametros[j]][i]=resultado1[i][j]
```

```
VtrendTN90[parametros[j]][i]=resultado2[i][j]
     OtrendTN90[parametros[j]][i]=resultado3[i][j]
     ItrendTN90[parametros[j]][i]=resultado4[i][j]
#TN10
resultado1= PTN10.apply(regresionlineal)
resultado2= VTN10.apply(regresionlineal)
resultado3 = OTN10.apply(regresionlineal)
resultado4= ITN10.apply(regresionlineal)
for i in range(0,len(nombre)):
  for j in range(0,len(parametros)):
     PtrendTN10[parametros[j]][i]=resultado1[i][j]
     VtrendTN10[parametros[j]][i]=resultado2[i][j]
     OtrendTN10[parametros[j]][i]=resultado3[i][j]
     ItrendTN10[parametros[j]][i]=resultado4[i][j]
#TXx
resultado1= PTXx.apply(regresionlineal)
resultado2= VTXx.apply(regresionlineal)
resultado3= OTXx.apply(regresionlineal)
resultado4= ITXx.apply(regresionlineal)
for i in range(0,len(nombre)):
  for j in range(0,len(parametros)):
     PtrendTXx[parametros[j]][i]=resultado1[i][j]
     VtrendTXx[parametros[j]][i]=resultado2[i][j]
     OtrendTXx[parametros[j]][i]=resultado3[i][j]
     ItrendTXx[parametros[j]][i]=resultado4[i][j]
#TXn
resultado1= PTXn.apply(regresionlineal)
resultado2= VTXn.apply(regresionlineal)
resultado3= OTXn.apply(regresionlineal)
resultado4= ITXn.apply(regresionlineal)
for i in range(0,len(nombre)):
  for j in range(0,len(parametros)):
     PtrendTXn[parametros[j]][i]=resultado1[i][j]
     VtrendTXn[parametros[j]][i]=resultado2[i][j]
     OtrendTXn[parametros[i]][i]=resultado3[i][j]
```

ItrendTXn[parametros[j]][i]=resultado4[i][j]

```
#TNx
```

resultado1= PTNx.apply(regresionlineal)
resultado2= VTNx.apply(regresionlineal)
resultado3= OTNx.apply(regresionlineal)
resultado4= ITNx.apply(regresionlineal)
for i in range(0,len(nombre)):
 for j in range(0,len(parametros)):
 PtrendTNx[parametros[j]][i]=resultado1[i][j]
 VtrendTNx[parametros[j]][i]=resultado3[i][j]
 ItrendTNx[parametros[j]][i]=resultado4[i][j]

#### #TNn

resultado1= PTNn.apply(regresionlineal)
resultado2= VTNn.apply(regresionlineal)
resultado3= OTNn.apply(regresionlineal)
resultado4= ITNn.apply(regresionlineal)
for i in range(0,len(nombre)):
 for j in range(0,len(parametros)):
 PtrendTNn[parametros[j]][i]=resultado1[i][j]
 VtrendTNn[parametros[j]][i]=resultado2[i][j]
 OtrendTNn[parametros[j]][i]=resultado4[i][j]
 ItrendTNn[parametros[j]][i]=resultado4[i][j]

## #FD

resultado2= VFD.apply(regresionlineal)
resultado3= OFD.apply(regresionlineal)
resultado4= IFD.apply(regresionlineal)
for i in range(0,len(nombre)):
 for j in range(0,len(parametros)):
 PtrendFD[parametros[j]][i]=resultado1[i][j]
 VtrendFD[parametros[j]][i]=resultado2[i][j]
 OtrendFD[parametros[j]][i]=resultado3[i][j]
 ItrendFD[parametros[j]][i]=resultado4[i][j]

resultado1= PFD.apply(regresionlineal)

# #ID

resultado1= PID.apply(regresionlineal) resultado2= VID.apply(regresionlineal) resultado3= OID.apply(regresionlineal)

```
resultado4= IID.apply(regresionlineal)
for i in range(0,len(nombre)):
  for j in range(0,len(parametros)):
     PtrendID[parametros[j]][i]=resultado1[i][j]
     VtrendID[parametros[j]][i]=resultado2[i][j]
     OtrendID[parametros[j]][i]=resultado3[i][j]
     ItrendID[parametros[j]][i]=resultado4[i][j]
# SU
resultado1= PSU.apply(regresionlineal)
resultado2= VSU.apply(regresionlineal)
resultado3 = OSU.apply(regresionlineal)
resultado4= ISU.apply(regresionlineal)
for i in range(0,len(nombre)):
  for j in range(0,len(parametros)):
     PtrendSU[parametros[j]][i]=resultado1[i][j]
     VtrendSU[parametros[j]][i]=resultado2[i][j]
     OtrendSU[parametros[j]][i]=resultado3[i][j]
     ItrendSU[parametros[j]][i]=resultado4[i][j]
#TR
resultado1= PTR.apply(regresionlineal)
resultado2= VTR.apply(regresionlineal)
resultado3 = OTR.apply(regresionlineal)
resultado4= ITR.apply(regresionlineal)
for i in range(0,len(nombre)):
  for j in range(0,len(parametros)):
     PtrendTR[parametros[j]][i]=resultado1[i][j]
     VtrendTR[parametros[j]][i]=resultado2[i][j]
     OtrendTR[parametros[j]][i]=resultado3[i][j]
     ItrendTR[parametros[j]][i]=resultado4[i][j]
#WSDI
resultado1= PWSDI.apply(regresionlineal)
resultado2= VWSDI.apply(regresionlineal)
resultado3 = OWSDI.apply(regresionlineal)
resultado4= IWSDI.apply(regresionlineal)
for i in range(0,len(nombre)):
 for j in range(0,len(parametros)):
```

```
PtrendWSDI[parametros[j]][i]=resultado1[i][j]
VtrendWSDI[parametros[j]][i]=resultado2[i][j]
OtrendWSDI[parametros[j]][i]=resultado3[i][j]
ItrendWSDI[parametros[j]][i]=resultado4[i][j]
```

### # CSDI

resultado1= PCSDI.apply(regresionlineal)
resultado2= VCSDI.apply(regresionlineal)
resultado3= OCSDI.apply(regresionlineal)
resultado4= ICSDI.apply(regresionlineal)
for i in range(0,len(nombre)):
 for j in range(0,len(parametros)):
 PtrendCSDI[parametros[j]][i]=resultado1[i][j]

VtrendCSDI[parametros[j]][i]=resultado2[i][j]
OtrendCSDI[parametros[j]][i]=resultado3[i][j]
ItrendCSDI[parametros[j]][i]=resultado4[i][j]

### # DTR

resultado1= PDTR.apply(regresionlineal)
resultado2= VDTR.apply(regresionlineal)
resultado3= ODTR.apply(regresionlineal)
resultado4= IDTR.apply(regresionlineal)
for i in range(0,len(nombre)):

for j in range(0,len(parametros)):

PtrendDTR[parametros[j]][i]=resultado1[i][j] VtrendDTR[parametros[j]][i]=resultado2[i][j] OtrendDTR[parametros[j]][i]=resultado3[i][j] ItrendDTR[parametros[j]][i]=resultado4[i][j]

#### #ETR

resultado1= PETR.apply(regresionlineal)
resultado2= VETR.apply(regresionlineal)
resultado3= OETR.apply(regresionlineal)
resultado4= IETR.apply(regresionlineal)
for i in range(0,len(nombre)):

for j in range(0,len(parametros)):

PtrendETR[parametros[j]][i]=resultado1[i][j] VtrendETR[parametros[j]][i]=resultado2[i][j] OtrendETR[parametros[j]][i]=resultado3[i][j] ItrendETR[parametros[j]][i]=resultado4[i][j]

```
# TEST MANN-KENDALL (Schramm, 2018)
from scipy.stats import norm
def mk_test(x, alpha=0.05):
  n = len(x)
  s = 0
  for k in range(n-1):
     for j in range(k+1, n):
       s += np.sign(x[j] - x[k]) # Calcula S
  unique x = np.unique(x) # Calcula valores unicos
  g = len(unique x)
  if n == g:
     var s = (n^*(n-1)^*(2^*n+5))/18 \# calcula VAR(S) si no hay valores repetidos
  else:
     tp = np.zeros(unique_x.shape)
     for i in range(len(unique_x)):
       tp[i] = sum(x == unique_x[i]) #calcula el nº de veces que se repite cada valor "unico"
     var s = (n*(n-1)*(2*n+5) - np.sum(tp*(tp-1)*(2*tp+5)))/18 # si hay valores repetidos en la
serie
   if s > 0:
     z = (s - 1)/np.sqrt(var_s)
  elif s < 0:
     z = (s + 1)/np.sqrt(var_s)
  else: \# s == 0:
     z = 0
  p = 2*(1-norm.cdf(abs(z))) # calcula el p value
  h = abs(z) > norm.ppf(1-alpha/2)
  if (z < 0) and h:
     trend = 'Decreasing'
  elif (z > 0) and h:
     trend = 'Increasing'
  else:
     trend = 'no trend'
  return trend, p
# Funcion que aplica el Test de MannKendall a una serie de datos y devuelve el resultado
def trends(df,parametro):
  columna = []
  for i in range (0,len(nb)): # Crea 1 columna de nan con el mismo numero de filas que de
estaciones
     columna.append(np.nan)
```

```
df['p value MK'] = columna #crea una columna nueva
df['Trend MK'] = columna #crea una columna nueva
for i in range (0,len(nb)):
    datos = parametro[nb[i]][pd.notna(parametro[nb[i]])] #datos a los que se les aplica MK
    x = datos.tolist() #Crea lista de valores
    trend, p = mk_test(x,0.05)
    df['p value MK'][nb[i]] = p
    df['Trend MK'][nb[i]] = trend
return df
```

tablas = [PtrendTX90, VtrendTX90, OtrendTX90, ItrendTX90, PtrendTX10, VtrendTX10, OtrendTX10, ItrendTX10, PtrendTN90, VtrendTN90, OtrendTN90, ItrendTN90, PtrendTN10, VtrendTN10, OtrendTN10, ItrendTN10, PtrendTXx, VtrendTXx, OtrendTXx, ItrendTXx, PtrendTXn, VtrendTXn, OtrendTXn, ItrendTXn, PtrendTNx, VtrendTNx, OtrendTNx, ItrendTNx, PtrendTNn, VtrendTNn, OtrendTNn, ItrendTNn, PtrendFD, VtrendFD, OtrendFD, ItrendFD, PtrendID, VtrendID, OtrendID, ItrendID, PtrendSU, VtrendSU, OtrendSU, ItrendSU, PtrendTR, VtrendTR, OtrendTR, ItrendTR, PtrendWSDI, VtrendDTR, VtrendDTR, OtrendDTR, OtrendDTR, ItrendCSDI, ItrendCSDI, PtrendDTR, VtrendDTR, OtrendDTR, ItrendDTR, PtrendETR, VtrendETR, VtrendETR, ItrendETR]

indices = [PTX90, VTX90, OTX90, ITX90, PTX10, VTX10, OTX10, ITX10, PTN90, VTN90, OTN90, ITN90, PTN10, VTN10, OTN10, ITN10, PTXx, VTXx, OTXx, ITXx, PTXn, VTXn, OTXn, ITXn, PTNx, VTNx, OTNx, ITNx, PTNn, VTNn, OTNn, ITNn, PFD, VFD, OFD, IFD, PID, VID, OID, IID, PSU, VSU, OSU, ISU, PTR, VTR, OTR, ITR, PWSDI, VWSDI, OWSDI, IWSDI, PCSDI, VCSDI, OCSDI, ICSDI, PDTR, VDTR, ODTR, IDTR, PETR, VETR, OETR, IETR]

```
for i in range (0,len(tablas)):
    a = tablas[i]
    b = indices[i]
    trends(a,b)
```

# # Guardar datos

nombres = ['P\_trendTX90', 'V\_trendTX90', 'O\_trendTX90', 'I\_trendTX90', 'P\_trendTX10', 'V\_trendTX10', 'O\_trendTX10', 'I\_trendTX10', 'P\_trendTN90', 'V\_trendTN90', 'O\_trendTN90', 'I\_trendTN10', 'V\_trendTN10', 'V\_trendTN10', 'I\_trendTN10', 'P\_trendTXx', 'V\_trendTXx', 'O\_trendTXx', 'I\_trendTXx', 'P\_trendTXn', 'V\_trendTXn', 'O\_trendTXn', 'I\_trendTXn', 'P\_trendTNx', 'V\_trendTNx', 'V\_trendTNx', 'V\_trendTNx', 'V\_trendTNn', 'V\_trendTNn', 'V\_trendTNn', 'V\_trendTNn', 'P\_trendTNn', 'I\_trendTNn', 'V\_trendTNn', 'V\_trendTNn', 'I\_trendID', 'V\_trendSU', 'V\_trendSU', 'V\_trendSU', 'V\_trendSU', 'V\_trendWSDI', 'V\_trendWS

```
'O_trendWSDI', 'I_trendWSDI', 'P_trendCSDI', 'V_trendCSDI', 'O_trendCSDI', 'I_trendCSDI', 'P_trendDTR', 'V_trendDTR', 'O_trendDTR', 'I_trendDTR', 'P_trendETR', 'V_trendETR', 'O_trendETR', 'I_trendETR']

for i in range(0,len(tablas)):
    tablas[i].to_csv(nombres[i]+'.csv',sep=";")
```

# A.3 Cálculo anual de los Índices de precipitación extrema

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from scipy import stats
# DATOS PRECIPITACION
pp =
pd.read_csv('precipitacion_cor.csv',sep=";",parse_dates=[0],dayfirst=True,index_col=['Fecha'])
#Carga los datos
pp anual = pp.groupby((pd.Grouper(freq='Y'))).sum(min count=1) # Precipitacion anual
acumulada
pp anual.index = pp anual.index.year # Cambio el indice a años
dTot = pp.groupby((pd.Grouper(freq='Y'))).count() #Recuento de datos totales por año
dTot.index = dTot.index.year #cambio el indice a años
# INDICE ESTACIONES y AÑOS
nb = pp anual.columns #Lista nombres abreviados (ALLZ, ASP, etc.)
estaciones = ['Allariz','As Pontes', 'Coruña', 'Ferrol', 'Fornelos', 'Labacolla', 'Lugo', 'Obs.(Stgo)',
'Ordes', 'Ourense', 'Pontevedra', 'Rozas', 'Sarria', 'Vigo']
anhos = set(pp.index.year) #Años
ind = [] # indice con los años
for i in anhos:
  ind.append(i)
# Tablas donde almacenar los datos
SDII = pd.DataFrame(index=pp anual.index,columns=nb)
CDD = pd.DataFrame(index=pp anual.index,columns=nb)
```

```
CWD = pd.DataFrame(index=pp_anual.index,columns=nb)
R10 = pd.DataFrame(index=pp anual.index,columns=nb)
R20 = pd.DataFrame(index=pp_anual.index,columns=nb)
R25 = pd.DataFrame(index=pp anual.index,columns=nb)
RX1D = pd.DataFrame(index=pp_anual.index,columns=nb)
RX5D = pd.DataFrame(index=pp anual.index,columns=nb)
R90p = pd.DataFrame(index=pp_anual.index,columns=nb)
R95p = pd.DataFrame(index=pp_anual.index,columns=nb)
R99p = pd.DataFrame(index=pp anual.index,columns=nb)
R95pTot = pd.DataFrame(index=pp_anual.index,columns=nb)
wet days = pd.DataFrame(index=pp anual.index,columns=nb)
wet_days_95p = pd.DataFrame(index=pp_anual.index,columns=nb)
# ÍNDICES: R10, R20, R25, RX1D
for i in range (0,len(nb)):
  y = 1960
  for j in range (0,len(ind)):
    condicion=((dTot[nb[i]][dTot.index==y]) >= 300).bool() #Excluyo años con menos de 300
días
    if condicion==True:
       a = pp[nb[i]][pp[nb[i]].index.year==y][pp[nb[i]]>=10] #días de ppt>=10
       R10[nb[i]][R10.index==y] = a.count()
       b= pp[nb[i]][pp[nb[i]].index.year==y][pp[nb[i]]>=20] #días de ppt>=20
       R20[nb[i]][R20.index==y] = b.count()
       c= pp[nb[i]][pp[nb[i]].index.year==y][pp[nb[i]]>=25] #días de ppt>=25
       R25[nb[i]][R20.index==y] = c.count()
       d= pp[nb[i]][pp[nb[i]].index.year==y].max() #ppt max acumulada en 1 dia
       RX1D[nb[i]][RX1D.index==y] = d
       y + = 1
    else:
       y+=1
# CALCULO PERCENTILES 90, 95 y 99 en el PERIODO DE REFERENCIA 1981-2010
p90 = []
p95 = []
p99 = []
for i in nb:
  inicio = 1981
  fin = 2010
  datos = pp[i][(pp.index.year>=inicio) & (pp.index.year<=fin)][pp[i]>=1]
```

```
p90.append(round(datos.quantile(0.90,),2))
  p95.append(round(datos.quantile(0.95,),2))
  p99.append(round(datos.quantile(0.99,),2))
# R90p, R95p, R99p
for i in range (0,len(nb)):
  for j in range (0,len(ind)):
     condicion=((dTot[nb[i]][dTot.index==ind[j]]) >= 300).bool()
     if condicion==True:
       datos = pp[nb[i]][pp[nb[i]].index.year==ind[j]][pp[nb[i]]>=1]
       ppt R90p = datos[datos>=p90[i]].sum() #Ppt superior al percentil 90 (periodo ref 1981-
2010)
       ppt_R95p = datos[datos>=p95[i]].sum() #Ppt superior al percentil 95 (periodo ref 1981-
2010)
       ppt_R99p = datos[datos>=p99[i]].sum() #Ppt superior al percentil 99 (periodo ref 1981-
2010)
       R90p[nb[i]][R90p.index==ind[j]] = ppt_R90p
       R95p[nb[i]][R95p.index==ind[j]] = ppt_R95p
       R99p[nb[i]][R99p.index==ind[j]] = ppt_R99p
#R95pTot
for i in range (0,len(nb)):
  for j in range (0,len(ind)):
     condicion=((dTot[nb[i]][dTot.index==ind[j]]) >= 300).bool()
     if condicion==True:
       datos = pp[nb[i]][pp[nb[i]].index.year==ind[j]][pp[nb[i]]>=1]
       ppt_p95 = datos[datos>=p95[i]].sum()
       R95pTot[nb[i]][R95p.index==ind[i]] =
(ppt_p95*100)/(pp_anual[nb[i]][pp_anual[nb[i]].index==ind[j]])
#SDII
wd = pd.DataFrame(index=ind, columns=nb)
for i in nb:
  for j in ind:
     condicion=((dTot[i][dTot.index==j]) >= 300).bool()
     if condicion==True:
       wd[i][wd.index==j] = pp[i][pp.index.year==j][pp[i]>=1].count() #no dias de ppt>=1
SDII = pp_anual/wd
# PRECIPITACION MÁXIMA ACUMULADA EN 5 DÍAS (RX5D)
```

```
RX5D = pd.DataFrame(index=pp_anual.index,columns=nb)
for i in range (0,len(nb)):
  y = 1960
  for j in range (0,len(ind)):
     condicion=((dTot[nb[i]][dTot.index==ind[j]]) >= 300).bool()
     if condicion==True:
       dias=[]
       I0 = pp[nb[i]][pp.index.year==y] #Lista de fecha y su dato correspondiente
       for d in range(0,len(l0.index)):
          dias.append(d)
       I0.index = dias #Cambio indice de fecha a int (1 dia por dato)
       maximo=0 #Precipitacion maxima acumulada
       dia=0 #Recorr posicion en la lista
       while dia<=(len(dias)-5): #recorre la serie de fechas hasta el día n-5.
          cont=0 #Cuenta nº dias sumados
          suma=0 #Cantidad de precipitacion
          nn=dia #Posicion en el periodo de 5 dias
          while cont<5:
            suma = suma + I0[nn]
            if suma>maximo:
              maximo = suma
            cont+=1 #cuenta dias sumados
            nn+=1 #Siguiente dato de la serie de 5
          dia+=1 #dia siguiente de la lista I0
       RX5D[nb[i]][RX5D.index==y] = maximo #Almaceno valor máximo de precipitacion
acumulada
     y += 1
# DIAS DE SEQUÍA CONSECUTIVOS (CDD)
CDD = pd.DataFrame(index=pp_anual.index,columns=nb)
for i in range (0,len(nb)):
  y = 1960
  for j in range (0,len(pp_anual.index)):
     condicion=((dTot[nb[i]][dTot.index==ind[j]]) >= 300).bool() # Excluyo años con menos de
300 datos
     if condicion==True:
       I0 = pp[nb[i]][pp.index.year==y][pp[nb[i]]<1].index #Lista fehcas donde pp<1 (dias secos)
       m= len(I0) #Longitud de la lista
       I1 = I0[0:m-1] #Nueva lista 1 = lista original desde la posicion 0 a la penúltima
       I2 = I0[1:m] #Nueva lista 2 = lista original desde la posicion 1 a la última
```

```
dl = (l2 - l1).days # Diferencia entre dias consecutivos en la lista
       long = len(dl) #Longitud de la lista de "dias de difrencia"
       cont = 0 #Contador de dias consecutivos de sequía
       j=0 #posicion inicial en la lista de dias
       ICDD = 0 #Almaceno serie mas larga de dias consecutivos de sequia
       for j in range (0,len(dl)-1):
          if dl[j]==1:
            cont+=1
            if cont>ICDD and dl[j+1]!=1:
               ICDD=cont
               cont=0
          else:
            cont=0
       CDD[nb[i]][CDD.index==y] = ICDD
     y+=1
# DIAS DE PRECIPITACION CONSECUTIVOS (CWD)
CWD = pd.DataFrame(index=pp_anual.index,columns=nb)
for i in range (0,len(nb)):
  y = 1960
  for j in range (0,len(pp_anual.index)):
     condicion=((dTot[nb[i]][dTot.index==ind[j]]) >= 300).bool()
     if condicion==True:
       10 = pp[nb[i]][pp.index.year==y][pp[nb[i]]>=1].index #Lista fechas donde ppt>=1 (dias
humedos)
       m = len(l0) #Longitud de la lista
       I1 = I0[0:m-1] #Nueva lista 1 = lista original desde la posicion 0 a la penúltima
       I2 = I0[1:m] #Nueva lista 2 = lista original desde la posicion 1 a la última
       dl = (l2 - l1).days # Diferencia entre dias consecutivos en la lista
       long = len(dl) #Longitud de la lista de días-difrencia
       cont = 0 #Contador de dias consecutivos de Iluvia
       ICWD = 0 #Almaceno serie mas larga de dias consecutivos de lluvia
       j=0 #posicion inicial en la lista de dias
       for j in range (0,long-1):
          if dl[i]==1:
            cont+=1
            if cont>ICWD and dl[j+1]!=1:
               ICWD=cont
               cont=0
          else:
```

```
cont=0
       CWD[nb[i]][CWD.index==y] = ICWD
    y+=1
# Guardar datos de las regresiones lineales
parametros = ['Slope', 'r value', 'p value', 'std error']
IrPPt = pd.DataFrame(index=nb,columns=parametros)
IrSDII = pd.DataFrame(index=nb,columns=parametros)
lrCDD = pd.DataFrame(index=nb,columns=parametros)
lrCWD = pd.DataFrame(index=nb,columns=parametros)
IrR10 = pd.DataFrame(index=nb,columns=parametros)
lrR20 = pd.DataFrame(index=nb,columns=parametros)
lrR25 = pd.DataFrame(index=nb,columns=parametros)
IrRX1D = pd.DataFrame(index=nb,columns=parametros)
IrRX5D = pd.DataFrame(index=nb,columns=parametros)
lrR90p = pd.DataFrame(index=nb,columns=parametros)
lrR95p = pd.DataFrame(index=nb,columns=parametros)
lrR99p = pd.DataFrame(index=nb,columns=parametros)
IrR95pTot = pd.DataFrame(index=nb,columns=parametros)
lrwet days = pd.DataFrame(index=nb,columns=parametros)
# CALCULA REGRESION LINEAL
def regresionlineal(u):
  u = u[\sim u.isnull()]
  x= u.index
  y= u.tolist()
  slope, intercept, r_value, p_value, std_err = stats.linregress(x,y)
  plt.plot(x, intercept + slope*x, 'r')
  return slope,r_value,p_value,std_err
# INDICE PrecTot
for i in range(0,len(nb)):
  fig=plt.figure(figsize=[10,5])
  f = pp_anual[dTot>=300][nb[i]].plot(style='.',title='Precipitación Total acumulada -
'+estaciones[i], color='b')
  f.set_ylabel('Precipitacion (mm)')
  f.set xlabel('')
# REGRESION LINEAL PrecTot
  u= pp_anual[dTot>=300][nb[i]]
```

```
resultado = regresionlineal(u)
  for n in range (0,len(parametros)):
     IrPPt[parametros[n]].loc[nb[i]]= resultado[n]
  plt.savefig('PrecTot - '+estaciones[i]+'.png')
# INDICE SDII
for i in range(0,len(nb)):
  fig=plt.figure(figsize=[10,5])
  f = SDII[dTot>=300][nb[i]].plot(style='.',title='SDII - '+estaciones[i],color='b')
  f.set_ylabel('Precipitacion (mm)')
  f.set_xlabel(' ')
# REGRESION LINEAL SDII
  u= SDII[dTot>=300][nb[i]]
  resultado = regresionlineal(u)
  for n in range (0,len(parametros)):
     IrSDII[parametros[n]].loc[nb[i]]= resultado[n]
  plt.savefig('SDII - '+estaciones[i]+'.png')
# INDICE CDD
for i in range(0,len(nb)):
  fig=plt.figure(figsize=[10,5])
  f = CDD[dTot>=300][nb[i]].plot(style='.',title='CDD - '+estaciones[i],color='b')
  f.set_ylabel('Precipitacion (mm)')
  f.set_xlabel(' ')
# REGRESION LINEAL CDD
  u= CDD[dTot>=300][nb[i]]
  resultado = regresionlineal(u)
  for n in range (0,len(parametros)):
     lrCDD[parametros[n]].loc[nb[i]]= resultado[n]
  plt.savefig('CDD - '+estaciones[i]+'.png')
# INDICE CWD
for i in range(0,len(nb)):
  fig=plt.figure(figsize=[10,5])
  f = CWD[dTot>=300][nb[i]].plot(style='.',title='CWD - '+estaciones[i],color='b')
  f.set_ylabel('Número maximo de días consecutivos de precipitación')
  f.set_xlabel(' ')
```

```
# REGRESION LINEAL CWD
  u = CWD[dTot >= 300][nb[i]]
  resultado = regresionlineal(u)
  for n in range (0,len(parametros)):
     lrCWD[parametros[n]].loc[nb[i]]= resultado[n]
  plt.savefig('CWD - '+estaciones[i]+'.png')
# INDICE R10
for i in range(0,len(nb)):
  fig=plt.figure(figsize=[10,5])
  f = R10[dTot>=300][nb[i]].plot(style='.',title=' - R10'+estaciones[i],color='b')
  f.set_ylabel('Precipitacion (mm)')
  f.set_xlabel(' ')
# REGRESION LINEAL R10
  u= R10[dTot>=300][nb[i]]
  resultado = regresionlineal(u)
  for n in range (0,len(parametros)):
     IrR10[parametros[n]].loc[nb[i]]= resultado[n]
  plt.savefig('R10 - '+estaciones[i]+'.png')
# INDICE R20
for i in range(0,len(nb)):
  fig=plt.figure(figsize=[10,5])
  f = R20[nb[i]].plot(style='.',title='R20 - '+estaciones[i],color='b')
  f.set_ylabel('Días')
  f.set_xlabel(' ')
# REGRESION LINEAL R20
  u= R20[dTot>=300][nb[i]]
  resultado = regresionlineal(u)
  for n in range (0,len(parametros)):
     lrR20[parametros[n]].loc[nb[i]]= resultado[n]
  plt.savefig('R20 - '+estaciones[i]+'.png')
# INDICE R25
for i in range(0,len(nb)):
  fig=plt.figure(figsize=[10,5])
  f = R25[dTot >= 300][nb[i]].plot(style='.',title='R25 - '+estaciones[i],color='b')
  f.set_ylabel('Precipitacion (mm)')
```

```
f.set_xlabel(' ')
# REGRESION LINEAL R25
  u= R25[dTot>=300][nb[i]]
  resultado = regresionlineal(u)
  for n in range (0,len(parametros)):
     IrR25[parametros[n]].loc[nb[i]]= resultado[n]
  plt.savefig('R25 - '+estaciones[i]+'.png')
# INDICE RX1D
for i in range(0,len(nb)):
  fig=plt.figure(figsize=[10,5])
  f = RX1D[dTot>=300][nb[i]].plot(style='.',title='RX1D - ' +estaciones[i],color='b')
  f.set_ylabel('Precipitacion (mm)')
  f.set_xlabel(' ')
# REGRESION LINEAL RX1D
  u = RX1D[dTot >= 300][nb[i]]
  resultado = regresionlineal(u)
  for n in range (0,len(parametros)):
     IrRX1D[parametros[n]].loc[nb[i]]= resultado[n]
  plt.savefig('RX1D - '+estaciones[i]+'.png')
# INDICE RX5D
for i in range(0,len(nb)):
  fig=plt.figure(figsize=[12,5])
  f = RX5D[dTot>=300][nb[i]].plot(style='.',title='RX5D - '+estaciones[i],color='b')
  f.set_ylabel('Precipitacion (mm)')
  f.set_xlabel(' ')
# REGRESION LINEAL RX5D
  u= RX5D[dTot>=300][nb[i]]
  resultado = regresionlineal(u)
  for n in range (0,len(parametros)):
     IrRX5D[parametros[n]].loc[nb[i]]= resultado[n]
  plt.savefig('RX5D - '+estaciones[i]+'.png')
# INDICE R90p
for i in range(0,len(nb)):
  fig=plt.figure(figsize=[10,5])
```

```
f = R90p[dTot>=300][nb[i]].plot(style='.',title='R90P - '+estaciones[i],color='b')
  f.set ylabel('Precipitacion (mm)')
  f.set_xlabel(' ')
# REGRESION LINEAL R90p
  u = R90p[dTot >= 300][nb[i]]
  resultado = regresionlineal(u)
  for n in range (0,len(parametros)):
     IrR90p[parametros[n]].loc[nb[i]]= resultado[n]
  plt.savefig('R90p - '+estaciones[i]+'.png')
# INDICE R95p
for i in range(0,len(nb)):
  fig=plt.figure(figsize=[10,5])
  f = R95p[dTot>=300][nb[i]].plot(style='.',title='R95P - '+estaciones[i],color='b')
  f.set_ylabel('Días')
  f.set_xlabel(' ')
# REGRESION LINEAL R95p
  u = R95p[dTot >= 300][nb[i]]
  resultado = regresionlineal(u)
  for n in range (0,len(parametros)):
     lrR95p[parametros[n]].loc[nb[i]]= resultado[n]
  plt.savefig('R95p - '+estaciones[i]+'.png')
# INDICE R99p
for i in range(0,len(nb)):
  fig=plt.figure(figsize=[10,5])
  f = R99p[dTot >= 300][nb[i]].plot(style='.',title='R99p - '+estaciones[i],color='b')
  f.set_ylabel('Días')
  #f.set_xlabel(' ')
# REGRESION LINEAL R99p
  u = R99p[dTot >= 300][nb[i]]
  resultado = regresionlineal(u)
  for n in range (0,len(parametros)):
     lrR99p[parametros[n]].loc[nb[i]]= resultado[n]
  #plt.savefig('R99p - '+estaciones[i]+'.png')
# INDICE R95pTot
```

```
for i in range(0,len(nb)):
  fig=plt.figure(figsize=[20,5])
  f = R95pTot[dTot>=300][nb[i]].plot(style='.',title='R95PTot - '+estaciones[i],color='b')
  f.set ylabel('%')
  f.set_xlabel(' ')
# REGRESION LINEAL (R95pTot)
  u = R95pTot[dTot>=300][nb[i]]
  resultado = regresionlineal(u)
  for n in range (0,len(parametros)):
     IrR95pTot[parametros[n]].loc[nb[i]]= resultado[n]
  plt.savefig('R95pTot - '+estaciones[i]+'.png')
# TEST MANN-KENDALL (Schramm, 2018)
from scipy.stats import norm
def mk test(x, alpha=0.05):
  n = len(x)
  s = 0
  for k in range(n-1):
     for j in range(k+1, n):
       s += np.sign(x[j] - x[k]) # Calcula S
  unique_x = np.unique(x) # Calcula valores unicos
  g = len(unique x)
  if n == g:
     var s = (n*(n-1)*(2*n+5))/18 \# calcula VAR(S) si no hay valores repetidos
  else:
     tp = np.zeros(unique_x.shape)
     for i in range(len(unique_x)):
       tp[i] = sum(x == unique_x[i]) #calcula el nº de veces que se repite cada valor "unico"
     var_s = (n*(n-1)*(2*n+5) - np.sum(tp*(tp-1)*(2*tp+5)))/18 # si hay valores repetidos en la
serie
   if s > 0:
    z = (s - 1)/np.sqrt(var_s)
  elif s < 0:
     z = (s + 1)/np.sqrt(var_s)
  else: # s == 0:
     z = 0
  p = 2*(1-norm.cdf(abs(z))) # calcula el p value
  h = abs(z) > norm.ppf(1-alpha/2)
  if (z < 0) and h:
```

```
trend = 'Decreasing'
  elif (z > 0) and h:
     trend = 'Increasing'
  else:
     trend = 'no trend'
  return trend, p
# Funcion que aplica el Test de MannKendall a una serie de datos y devuelve el resultado
def trends(df,parametro):
  columna = []
  for i in range (0,len(nb)): # Crea 1 columna de nan con el mismo numero de filas que de
estaciones
     columna.append(np.nan)
  df['p value MK'] = columna #crea una columna nueva
  df['Trend MK'] = columna #crea una columna nueva
  for i in range (0,len(nb)):
     datos = parametro[nb[i]][pd.notna(parametro[nb[i]])] #datos a los que se les aplica MK
    x = datos.tolist() #Crea lista de valores
     trend, p = mk_t(x,0.05)
    df['p value MK'][nb[i]] = p
     df['Trend MK'][nb[i]] = trend
  return df
tablas = [IrPPt, IrSDII, IrCDD, IrCWD, IrR10, IrR20, IrR25, IrRX1D, IrRX5D, IrR90p, IrR95p, IrR99p,
IrR95pTot]
indices = [pp_anual, SDII, CDD, CWD, R10, R20, R25, RX1D, RX5D, R90p, R95p, R99p,
R95pTot]
for i in range (0, len(tablas)):
  a = tablas[i]
  b = indices[i]
  trends(a,b)
# Guarda las df en formato .csv
IrPPt.to_csv('A_PPt.csv',sep=";")
IrSDII.to_csv('A_SDII.csv',sep=";")
IrCDD.to csv('A CDD.csv',sep=";")
IrCWD.to_csv('A_CWD.csv',sep=";")
IrR10.to_csv('A_R10.csv',sep=";")
IrR20.to csv('A R20.csv',sep=";")
IrR25.to_csv('A_R25.csv',sep=";")
IrRX1D.to_csv('A_RX1D.csv',sep=";")
```

```
IrRX5D.to_csv('A_RX5D.csv',sep=";")
IrR90p.to_csv('A_R90p.csv',sep=";")
IrR95p.to_csv('A_R95p.csv',sep=";")
IrR99p.to_csv('A_R99p.csv',sep=";")
IrR95pTot.to_csv('A_R95pTot.csv',sep=";")
```

# A.4 Cálculo estacional de los Índices de precipitación extrema

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from scipy import stats
# DATOS PRECIPITACION
pp =
pd.read_csv('precipitacion_cor.csv',sep=";",parse_dates=[0],dayfirst=True,index_col=['Fecha'])
# NOMBRE ESTACIONES
nombre = pp.columns
estaciones = ['Allariz', 'As Pontes', 'Coruña', 'Ferrol', 'Fornelos', 'Labacolla', 'Lugo', 'Obs.(Stgo)',
'Ordes', 'Ourense', 'Pontevedra', 'Rozas', 'Sarria', 'Vigo']
# INDICE AÑOS
anhos = set(pp.index.year)
ind = []
for i in anhos:
  ind.append(i)
indices =
['PT','wd','SDII','CDD','CWD','R10','R20','R25','RX1D','RX5D','R90p','R95p','R99p','R95pTot']
listatablas= []
for i in indices: # Crea tablas donde almacenar resultados del calculo de los indices
  listatablas.append('P'+str(i))
  listatablas.append('V'+str(i))
  listatablas.append('O'+str(i))
  listatablas.append('I'+str(i))
for i in listatablas:
```

```
#FUNCIONES
def PrecTot(trimestre):
  n_nan = trimestre.isnull().sum()
  if n_nan<=15:
     PT = trimestre.sum()
  else:
     PT = np.nan
  return PT
def wet_days(trimestre):
  n_nan = trimestre.isnull().sum()
  if n_nan<=15:
     num = trimestre[trimestre>=1].count()
  else:
     num = np.nan
  return num
def R10(trimestre):
  n_nan = trimestre.isnull().sum()
  if n_nan<=15:
     R10 = trimestre[trimestre>=10].count()
  else:
     R10 = np.nan
  return R10
def R20(trimestre):
  n_nan = trimestre.isnull().sum()
  if n_nan<=15:
     R20 = trimestre[trimestre>=20].count()
  else:
     R20 = np.nan
  return R20
def R25(trimestre):
  n_nan = trimestre.isnull().sum()
  if n_nan<=15:
```

R25 = trimestre[trimestre>=25].count()

else:

R25 = np.nan

```
return R25
```

```
def RX1D(trimestre):
  n_nan = trimestre.isnull().sum()
  if n_nan<=15:
     RX1D = trimestre.max()
  else:
     RX1D = np.nan
  return RX1D
def RX5D(trimestre):
  n_nan = trimestre.isnull().sum()
  if n_nan<=15:
    trimestre.index = list(range(0,len(trimestre.index)))
     RX5D = periodo5d(trimestre)
  else:
     RX5D = np.nan
  return RX5D
def periodo5d(trimestre):
  d=0
  dia = 0
  maximo=0 #Guardo ppt maxima acumulada
  I = len(trimestre.index)
  while dia<=(I-5): #recorre la serie de fechas
     suma=trimestre[dia:dia+4].sum()
     if suma>maximo:
       maximo = suma
    dia+=1
  return máximo
def nulos(trimestre):
  n_nan = trimestre.isnull().sum()
  return n_nan
def seleccion(datos):
  lista = datos[datos>=1].index
  return lista
def seleccion2(datos):
  lista = datos[datos<1].index
```

### return lista

```
def CWD(datos):
  lista = seleccion(datos)
  n_nan = nulos(datos)
  if n_nan<=15:
    I = len(lista)
    dif = (lista[1:l]-lista[0:(l-1)]).days
    dif= dif.tolist()
    CWD = SerieMaxWd(dif)
    return CWD
  else:
     return np.nan
def SerieMaxWd(dif):
  valorCWD=0
  cont=0
  long = len(dif)
  for d in range (0,long-1):
     if dif[d]==1:
       cont+=1
       if (cont>valorCWD) and (dif[d+1]!=1):
          valorCWD = cont
          cont=0
     else:
       cont=0
  return valorCWD
def CDD(datos):
  lista = seleccion2(datos)
  n_nan = nulos(datos)
  if n_nan<=15:
    I = len(lista)
    dif = (lista[1:l]-lista[0:(l-1)]).days
    dif= dif.tolist()
    CDD = SerieMaxDD(dif)
    return CDD
  else:
     return np.nan
```

```
def SerieMaxDD(dif):
  valorCDD=0
  cont=0
  long = len(dif)
  for d in range (0,long-1):
     if dif[d]==1:
       cont+=1
       if (cont>valorCDD) and (dif[d+1]!=1):
          valorCDD = cont
          cont=0
     else:
       cont=0
  return valorCDD
# CALCULO INDICES
# INDICE CWD - PVO
y = 1960
f = y + len(ind)
while y<f:
  P = pp[((pp.index.month >= 3) & (pp.index.month <= 5)) & (pp.index.year == y)]
  V = pp[((pp.index.month \ge 6) \& (pp.index.month \le 8)) \& (pp.index.year == y)]
  O = pp[((pp.index.month \ge 9) & (pp.index.month \le 11)) & (pp.index.year == y)]
  valorP = P.apply(CWD)
  valorV = V.apply(CWD)
  valorO = O.apply(CWD)
  for i in range(0,len(valorP)):
     PCWD[nombre[i]][PCWD.index==y] = valorP[i]
     VCWD[nombre[i]][VCWD.index==y] = valorV[i]
     OCWD[nombre[i]][OCWD.index==y] = valorO[i]
  y+=1
# INDICE CWD - I
y = 1961
f = y + len(ind)
while y<f-1:
  I = pp[( ((pp.index.month==12) & (pp.index.year == y-1)) | ((pp.index.month<=2) &
  (pp.index.month>=1) & (pp.index.year==y)) )]
  valor = I.apply(CWD)
  for i in range(0,len(valor)):
```

```
ICWD[nombre[i]][ICWD.index==y] = valor[i]
      y+=1
# INDICE CDD - PVO
y = 1960
f = y + len(ind)
while y<f:
       P = pp[((pp.index.month \ge 3) \& (pp.index.month \le 5)) \& (pp.index.year == y)]
       V = pp[((pp.index.month \ge 6) \& (pp.index.month \le 8)) \& (pp.index.year == y)]
       O = pp[((pp.index.month \ge 9) \& (pp.index.month \le 11)) \& (pp.index.year == y)]
       valorP = P.apply(CDD)
       valorV = V.apply(CDD)
       valorO = O.apply(CDD)
       for i in range(0,len(valorP)):
             PCDD[nombre[i]][PCDD.index==y] = valorP[i]
             VCDD[nombre[i]][VCDD.index==y] = valorV[i]
             OCDD[nombre[i]][OCDD.index==y] = valorO[i]
       y + = 1
# INDICE CDD - I
y = 1961
f = y + len(ind)
while y<f-1:
       I = pp[((pp.index.month==12) & (pp.index.year == y-1)) | ((pp.index.month<=2) & (pp.index.month==12) & (pp.index.year == y-1)) | ((pp.index.month==12) & (pp.index.month==12) & (pp.index.year == y-1)) | ((pp.index.month==12) & (pp.index.year == y-1)) | ((pp.index.month==12) & (pp.index.year == y-1)) | ((pp.index.month==12) & (pp.index.month==12) & (pp.index.month
(pp.index.month>=1) & (pp.index.year==y)) )]
       valor = I.apply(CDD)
      for i in range(0,len(valor)):
             ICDD[nombre[i]][ICDD.index==y] = valor[i]
      y+=1
# INDICE PrecTot - PVO
y = 1960
f = y + len(ind)
while y<f:
       P = pp[((pp.index.month >= 3) & (pp.index.month <= 5)) & (pp.index.year == y)]
       V = pp[((pp.index.month >= 6) & (pp.index.month <= 8)) & (pp.index.year == y)]
       O = pp[((pp.index.month \ge 9) & (pp.index.month \le 11)) & (pp.index.year == y)]
       P = P.apply(PrecTot)
       V = V.apply(PrecTot)
       O = O.apply(PrecTot)
```

```
for i in range(0,len(P)):
              PPT[nombre[i]][PPT.index==y] = P[i]
              VPT[nombre[i]][VPT.index==y] = V[i]
              OPT[nombre[i]][OPT.index==y] = O[i]
       y+=1
# INDICE PrecTot - I
y = 1961
f = y + len(ind)
while y<f-1:
       I = pp[((pp.index.month==12) & (pp.index.year == y-1)) | ((pp.index.month<=2) & (pp.index.month==12) & (pp.index.year == y-1)) | ((pp.index.month==12) & (pp.index.month==12) & (pp.index.year == y-1)) | ((pp.index.month==12) & (pp.index.year == y-1)) | ((pp.index.month==12) & (pp.index.year == y-1)) | ((pp.index.month==12) & (pp.index.month==12) & (pp.index.month
(pp.index.month>=1) & (pp.index.year==y)) )]
       I = I.apply(PrecTot)
       for i in range(0,len(I)):
              IPT[nombre[i]][IPT.index==y] = I[i]
       y+=1
#WET DAYS - PVO
y = 1960
f = y + len(ind)
while y<f:
       P = pp[((pp.index.month >= 3) & (pp.index.month <= 5)) & (pp.index.year == y)]
       V = pp[((pp.index.month \ge 6) \& (pp.index.month \le 8)) \& (pp.index.year == y)]
       O = pp[((pp.index.month \ge 9) \& (pp.index.month \le 11)) \& (pp.index.year == y)]
       P = P.apply(wet_days)
       V = V.apply(wet_days)
       O = O.apply(wet_days)
       for i in range(0,len(P)):
              Pwd[nombre[i]][Pwd.index==y] = P[i]
              Vwd[nombre[i]][Vwd.index==y] = V[i]
              Owd[nombre[i]][Owd.index==y] = O[i]
       y+=1
#WET DAYS - I
y = 1961
f = y + len(ind)
while y<f-1:
       I = pp[( ((pp.index.month==12) & (pp.index.year == y-1)) | ((pp.index.month<=2) &
(pp.index.month>=1) & (pp.index.year==y)) )]
       I = I.apply(wet_days)
```

```
for i in range(0,len(I)):
     Iwd[nombre[i]][Iwd.index==y] = I[i]
  y+=1
# INDICE SDII - PVOI
try:
  PSDII = PPT/Pwd
  VSDII = VPT/Vwd
  OSDII = OPT/Owd
  ISDII = IPT/Iwd
except:
  ZeroDivisionError
# INDICES R10, R20, R25 - PVO
y = 1960
f = y + len(ind)
while y<f:
  P = pp[((pp.index.month >= 3) \& (pp.index.month <= 5)) \& (pp.index.year == y)]
  V = pp[((pp.index.month >= 6) & (pp.index.month <= 8)) & (pp.index.year == y)]
  O = pp[((pp.index.month \ge 9) \& (pp.index.month \le 11)) \& (pp.index.year == y)]
  pR10 = P.apply(R10)
  vR10 = V.apply(R10)
  oR10 = O.apply(R10)
  pR20 = P.apply(R20)
  vR20 = V.apply(R20)
  oR20 = O.apply(R20)
  pR25 = P.apply(R25)
  vR25 = V.apply(R25)
  oR25 = O.apply(R25)
  for i in range(0,len(pR10)):
     PR10[nombre[i]][PR10.index==y] = pR10[i]
     PR20[nombre[i]][PR20.index==y] = pR20[i]
     PR25[nombre[i]][PR25.index==y] = pR25[i]
     VR10[nombre[i]][VR10.index==y] = vR10[i]
    VR20[nombre[i]][PR20.index==y] = vR20[i]
    VR25[nombre[i]][PR25.index==y] = vR25[i]
    OR10[nombre[i]][Owd.index==y] = oR10[i]
     OR20[nombre[i]][OR20.index==y] = oR20[i]
     OR25[nombre[i]][OR25.index==y] = oR25[i]
  y+=1
```

```
# INDICES R10 R20, R25 - I
y = 1961
f = y + len(ind)
while y<f-1:
           I = pp[((pp.index.month==12) & (pp.index.year == y-1)) | ((pp.index.month<=2) & (pp.index.month==12) & (pp.index.year == y-1)) | ((pp.index.month==12) & (pp.index.month==12) & (pp.index.year == y-1)) | ((pp.index.month==12) & (pp.index.year == y-1)) | ((pp.index.month==12) & (pp.index.year == y-1)) | ((pp.index.month==12) & (pp.index.month==12) & (pp.index.month
 (pp.index.month>=1) & (pp.index.year==y)) )]
           I10 = I.apply(R10)
           I20 = I.apply(R20)
           125 = I.apply(R25)
           for i in range(0,len(I10)):
                      IR10[nombre[i]][IR10.index==y] = I10[i]
                      IR20[nombre[i]][IR20.index==y] = I20[i]
                      IR25[nombre[i]][IR25.index==y] = I25[i]
           y+=1
# INDICE RX1D - PVO
y = 1960
f = y+len(ind)
while y<f:
           P = pp[((pp.index.month \ge 3) \& (pp.index.month \le 5)) \& (pp.index.year == y)]
           V = pp[((pp.index.month >= 6) & (pp.index.month <= 8)) & (pp.index.year == y)]
           O = pp[((pp.index.month \ge 9) & (pp.index.month \le 11)) & (pp.index.year == y)]
           P1D = P.apply(RX1D)
           V1D = V.apply(RX1D)
           O1D = O.apply(RX1D)
           for i in range(0,len(P1D)):
                      PRX1D[nombre[i]][PRX1D.index==y] = P1D[i]
                      VRX1D[nombre[i]][VRX1D.index==y] = V1D[i]
                      ORX1D[nombre[i]][ORX1D.index==y] = O1D[i]
           y+=1
# INDICE RX1D - I
y = 1961
f = y + len(ind)
while y<f-1:
           I = pp[((pp.index.month==12) & (pp.index.year == y-1)) | ((pp.index.month<=2) & (pp.index.month==12) & (pp.index.year == y-1)) | ((pp.index.month==12) & (pp.index.month==12) & (pp.index.year == y-1)) | ((pp.index.month==12) & (pp.index.year == y-1)) | ((pp.index.month==12) & (pp.index.year == y-1)) | ((pp.index.month==12) & (pp.index.month==12) & (pp.index.month
 (pp.index.month>=1) & (pp.index.year==y)) )]
           I1D = I.apply(RX1D)
           for i in range(0,len(I1D)):
```

```
IRX1D[nombre[i]][IRX1D.index==y] = I1D[i]
      y+=1
# INDICE RX5D - PVO
y = 1960
f = y + len(ind)
while y<f:
      P = pp[((pp.index.month >= 3) & (pp.index.month <= 5)) & (pp.index.year == y)]
      V = pp[((pp.index.month \ge 6) \& (pp.index.month \le 8)) \& (pp.index.year == y)]
      O = pp[((pp.index.month \ge 9) \& (pp.index.month \le 11)) \& (pp.index.year == y)]
      P5D = P.apply(RX5D)
      V5D = V.apply(RX5D)
      O5D = O.apply(RX5D)
      for i in range(0,len(P5D)):
            PRX5D[nombre[i]][PRX5D.index==y] = P5D[i]
           VRX5D[nombre[i]][VRX5D.index==y] = V5D[i]
            ORX5D[nombre[i]][ORX5D.index==y] = O5D[i]
      y + = 1
# INDICE RX5D - I
y = 1961
f = y + len(ind)
while y<f-1:
      I = pp[((pp.index.month==12) & (pp.index.year == y-1)) | ((pp.index.month<=2) & (pp.index.month==12) & (pp.index.year == y-1)) | ((pp.index.month==12) & (pp.index.month==12) & (pp.index.year == y-1)) | ((pp.index.month==12) & (pp.index.year == y-1)) | ((pp.index.month==12) & (pp.index.year == y-1)) | ((pp.index.month==12) & (pp.index.month==12) & (pp.index.month
(pp.index.month>=1) & (pp.index.year==y)) )]
      I5D = I.apply(RX5D)
     for i in range(0,len(I1D)):
            IRX5D[nombre[i]][IRX5D.index==y] = I5D[i]
      y+=1
# Percentiles 90, 95 y 99 periodo de referencia 1981-2010
tot_P = pp[pp >= 1][((pp.index.month >= 3) & (pp.index.month <= 5)) &
((pp.index.year>=1981)&(pp.index.year<=2010))]
tot_V = pp[pp >= 1][((pp.index.month >= 6) & (pp.index.month <= 8)) &
((pp.index.year>=1981)&(pp.index.year<=2010))]
tot_O = pp[pp \ge 1][(pp.index.month \ge 9) & (pp.index.month \le 11)) &
((pp.index.year>=1981)&(pp.index.year<=2010)) ]
tot I = pp[pp>=1][(pp.index.year>=1981) & (pp.index.year<=2010) & ((pp.index.month==12) |
 (pp.index.month==1) | (pp.index.month==2))]
Pp90 = tot_P.quantile(0.90,)
```

```
Vp90 = tot_V.quantile(0.90,)
Op90 = tot_O.quantile(0.90,)
lp90 = tot_l.quantile(0.90,)
Pp95 = tot_P.quantile(0.95,)
Vp95 = tot_V.quantile(0.95,)
Op95 = tot_O.quantile(0.95,)
lp95 = tot_I.quantile(0.95,)
Pp99 = tot_P.quantile(0.99,)
Vp99 = tot_V.quantile(0.99,)
Op99 = tot_O.quantile(0.99,)
lp99 = tot I.quantile(0.99,)
# INDICE R90p - PVO
y = 1960
f = y + len(ind)
while y<f:
  P = pp[((pp.index.month >= 3) & (pp.index.month <= 5)) & (pp.index.year == y)]
  V = pp[((pp.index.month >= 6) & (pp.index.month <= 8)) & (pp.index.year == y)]
  O = pp[((pp.index.month \ge 9) \& (pp.index.month \le 11)) \& (pp.index.year == y)]
  n_n = nulos(P)
  n_nanV = nulos(V)
  n_n = nulos(O)
  n_n = (n_n = 15)
  n_n = (n_n = 15)
  n_n = (n_n = 0)
  prim = P[P>=Pp90].sum()
  ver = V[V>=Vp90].sum()
  ot = O[O >= Op90].sum()
  for i in nombre:
     if n_nanP[i]!=True:
       prim[i] = np.nan
  for i in nombre:
     if n_nanV[i]!=True:
       ver[i] = np.nan
  for i in nombre:
     if n_nanO[i]!=True:
       ot[i] = np.nan
  for i in range(0,len(prim)):
     PR90p[nombre[i]][PR90p.index==y] = prim[i]
     VR90p[nombre[i]][VR90p.index==y] = ver[i]
```

```
OR90p[nombre[i]][OR90p.index==y] = ot[i]
  y+=1
# INDICE R95p - PVO
y = 1960
f = y + len(ind)
while y<f:
  P = pp[((pp.index.month \ge 3) \& (pp.index.month \le 5)) \& (pp.index.year == y)]
  V = pp[((pp.index.month \ge 6) \& (pp.index.month \le 8)) \& (pp.index.year == y)]
  O = pp[((pp.index.month \ge 9) \& (pp.index.month \le 11)) \& (pp.index.year == y)]
  n_n = nulos(P)
  n_n = nulos(V)
  n_n = nulos(O)
  n_n = (n_n = 15)
  n_n = (n_n = 15)
  n_nanO = (n_nanO<=15)
  prim = P[P>=Pp95].sum()
  ver = V[V>=Vp95].sum()
  ot = O[O >= Op95].sum()
  for i in nombre:
     if n_nanP[i]!=True:
       prim[i] = np.nan
  for i in nombre:
     if n_nanV[i]!=True:
       ver[i] = np.nan
  for i in nombre:
     if n_nanO[i]!=True:
       ot[i] = np.nan
  for i in range(0,len(prim)):
     PR95p[nombre[i]][PR95p.index==y] = prim[i]
     VR95p[nombre[i]][VR95p.index==y] = ver[i]
     OR95p[nombre[i]][OR95p.index==y] = ot[i]
  y+=1
# INDICE R99p - PVO
y = 1960
f = y + len(ind)
while y<f:
  P = pp[((pp.index.month >= 3) & (pp.index.month <= 5)) & (pp.index.year == y)]
  V = pp[((pp.index.month \ge 6) \& (pp.index.month \le 8)) \& (pp.index.year == y)]
```

```
O = pp[((pp.index.month \ge 9) & (pp.index.month \le 11)) & (pp.index.year == y)]
  n_nanP = nulos(P)
  n_nanV = nulos(V)
  n_n = nulos(O)
  n_n = (n_n = 15)
  n_n = (n_n = 15)
  n_nanO = (n_nanO<=15)
  prim = P[P>=Pp99].sum()
  ver = V[V >= Vp99].sum()
  ot = O[O >= Op99].sum()
  for i in nombre:
     if n_nanP[i]!=True:
       prim[i] = np.nan
  for i in nombre:
     if n_nanV[i]!=True:
       ver[i] = np.nan
  for i in nombre:
     if n_nanO[i]!=True:
       ot[i] = np.nan
  for i in range(0,len(prim)):
     PR99p[nombre[i]][PR99p.index==y] = prim[i]
     VR99p[nombre[i]][VR99p.index==y] = ver[i]
    OR99p[nombre[i]][OR99p.index==y] = ot[i]
  y+=1
# INDICES R90p, R95p, R99p - I
y = 1961
f = y + len(ind)
while y<f-1:
  I = pp[( ((pp.index.month==12) & (pp.index.year == y-1)) | ((pp.index.month<=2) &
(pp.index.month>=1) & (pp.index.year==y)) )]
  n_nanl = nulos(I)
  n_n = (n_n = 15)
  inv90 = I[I > = Ip90].sum()
  inv95 = I[I > = Ip95].sum()
  inv99 = I[I > = Ip99].sum()
  for i in nombre:
     if n_nanl[i]!=True:
       inv90[i] = np.nan
       inv95[i] = np.nan
```

```
inv99[i] = np.nan
  for i in range(0,len(nombre)):
     IR90p[nombre[i]][IR90p.index==y] = inv90[i]
     IR95p[nombre[i]][IR95p.index==y] = inv95[i]
     IR99p[nombre[i]][IR99p.index==y] = inv99[i]
  y+=1
# INDICE R95pTot - PVOI
PR95pTot = (PR95p*100)/PPT
VR95pTot = (VR95p*100)/VPT
OR95pTot = (OR95p*100)/OPT
IR95pTot = (IR95p*100)/IPT
# CALCULO TENDENCIAS
parametros = ['Slope', 'r value', 'p value', 'std error']
indices = ['trendPT', 'trendwd', 'trendSDII', 'trendCDD', 'trendCWD', 'trendR10', 'trendR20',
'trendR25', 'trendRX1D', 'trendRX5D', 'trendR90p', 'trendR95p', 'trendR99p', 'trendR95pTot']
listatablas= []
for i in indices:
  listatablas.append('P'+str(i))
  listatablas.append('V'+str(i))
  listatablas.append('O'+str(i))
  listatablas.append('I'+str(i))
for i in listatablas:
  exec('{}= pd.DataFrame(index=nombre, columns=parametros)'.format(i))
def regresionlineal(u):
  u = u[\sim u.isnull()]
  x= u.index
  y= u.tolist()
  slope, intercept, r_value, p_value, std_err = stats.linregress(x,y)
  return slope, r_value, p_value, std_err
#REGRESIONES LINEALES
# PRECIPITACION TOTAL ANUAL ACUMULADA
resultado1= PPT.apply(regresionlineal)
resultado2= VPT.apply(regresionlineal)
resultado3= OPT.apply(regresionlineal)
resultado4= IPT.apply(regresionlineal)
for i in range(0,len(nombre)):
```

```
for j in range(0,len(parametros)):
     PtrendPT[parametros[j]][i]=resultado1[i][j]
     VtrendPT[parametros[j]][i]=resultado2[i][j]
     OtrendPT[parametros[j]][i]=resultado3[i][j]
     ItrendPT[parametros[j]][i]=resultado4[i][j]
#SDII
resultado1= PSDII.apply(regresionlineal)
resultado2= VSDII.apply(regresionlineal)
resultado3 = OSDII.apply(regresionlineal)
resultado4= ISDII.apply(regresionlineal)
for i in range(0,len(nombre)):
  for j in range(0,len(parametros)):
     PtrendSDII[parametros[j]][i]=resultado1[i][j]
     VtrendSDII[parametros[j]][i]=resultado2[i][j]
     OtrendSDII[parametros[j]][i]=resultado3[i][j]
     ItrendSDII[parametros[j]][i]=resultado4[i][j]
#CDD
resultado1= PCDD.apply(regresionlineal)
resultado2= VCDD.apply(regresionlineal)
resultado3 = OCDD.apply(regresionlineal)
resultado4= ICDD.apply(regresionlineal)
for i in range(0,len(nombre)):
  for j in range(0,len(parametros)):
     PtrendCDD[parametros[j]][i]=resultado1[i][j]
     VtrendCDD[parametros[j]][i]=resultado2[i][j]
     OtrendCDD[parametros[j]][i]=resultado3[i][j]
     ItrendCDD[parametros[j]][i]=resultado4[i][j]
#CWD
resultado1= PCWD.apply(regresionlineal)
resultado2= VCWD.apply(regresionlineal)
resultado3= OCWD.apply(regresionlineal)
resultado4= ICWD.apply(regresionlineal)
for i in range(0,len(nombre)):
  for j in range(0,len(parametros)):
     PtrendCWD[parametros[j]][i]=resultado1[i][j]
```

VtrendCWD[parametros[j]][i]=resultado2[i][j]
OtrendCWD[parametros[j]][i]=resultado3[i][j]

# ItrendCWD[parametros[j]][i]=resultado4[i][j]

# #R10 resultado1= PR10.apply(regresionlineal) resultado2= VR10.apply(regresionlineal) resultado3 = OR10.apply(regresionlineal) resultado4= IR10.apply(regresionlineal) for i in range(0,len(nombre)): for j in range(0,len(parametros)): PtrendR10[parametros[j]][i]=resultado1[i][j] VtrendR10[parametros[j]][i]=resultado2[i][j] OtrendR10[parametros[j]][i]=resultado3[i][j] ItrendR10[parametros[j]][i]=resultado4[i][j] #R20 resultado1= PR20.apply(regresionlineal) resultado2= VR20.apply(regresionlineal) resultado3 = OR20.apply(regresionlineal) resultado4= IR20.apply(regresionlineal) for i in range(0,len(nombre)): for j in range(0,len(parametros)): PtrendR20[parametros[j]][i]=resultado1[i][j] VtrendR20[parametros[j]][i]=resultado2[i][j] OtrendR20[parametros[j]][i]=resultado3[i][j] ItrendR20[parametros[j]][i]=resultado4[i][j] #R25 resultado1= PR25.apply(regresionlineal) resultado2= VR25.apply(regresionlineal) resultado3 = OR25.apply(regresionlineal) resultado4= IR25.apply(regresionlineal) for i in range(0,len(nombre)): for j in range(0,len(parametros)): PtrendR25[parametros[j]][i]=resultado1[i][j] VtrendR25[parametros[j]][i]=resultado2[i][j] OtrendR25[parametros[j]][i]=resultado3[i][j] ItrendR25[parametros[j]][i]=resultado4[i][j] #RX1D

resultado1= PRX1D.apply(regresionlineal)

```
resultado2= VRX1D.apply(regresionlineal)
resultado3= ORX1D.apply(regresionlineal)
resultado4= IRX1D.apply(regresionlineal)
for i in range(0,len(nombre)):
  for j in range(0,len(parametros)):
     PtrendRX1D[parametros[j]][i]=resultado1[i][j]
     VtrendRX1D[parametros[j]][i]=resultado2[i][j]
     OtrendRX1D[parametros[j]][i]=resultado3[i][j]
     ItrendRX1D[parametros[j]][i]=resultado4[i][j]
#RX5D
resultado1= PRX5D.apply(regresionlineal)
resultado2= VRX5D.apply(regresionlineal)
resultado3= ORX5D.apply(regresionlineal)
resultado4= IRX5D.apply(regresionlineal)
for i in range(0,len(nombre)):
  for j in range(0,len(parametros)):
     PtrendRX5D[parametros[j]][i]=resultado1[i][j]
     VtrendRX5D[parametros[j]][i]=resultado2[i][j]
     OtrendRX5D[parametros[j]][i]=resultado3[i][j]
     ItrendRX5D[parametros[j]][i]=resultado4[i][j]
# R90P
resultado1= PR90p.apply(regresionlineal)
resultado2= VR90p.apply(regresionlineal)
resultado3= OR90p.apply(regresionlineal)
resultado4= IR90p.apply(regresionlineal)
for i in range(0,len(nombre)):
  for j in range(0,len(parametros)):
     PtrendR90p[parametros[j]][i]=resultado1[i][j]
     VtrendR90p[parametros[j]][i]=resultado2[i][j]
     OtrendR90p[parametros[j]][i]=resultado3[i][j]
     ItrendR90p[parametros[j]][i]=resultado4[i][j]
#R95p
```

resultado1= PR95p.apply(regresionlineal) resultado2= VR95p.apply(regresionlineal) resultado3= OR95p.apply(regresionlineal) resultado4= IR95p.apply(regresionlineal) for i in range(0,len(nombre)):

```
for j in range(0,len(parametros)):
     PtrendR95p[parametros[j]][i]=resultado1[i][j]
     VtrendR95p[parametros[j]][i]=resultado2[i][j]
     OtrendR95p[parametros[j]][i]=resultado3[i][j]
     ItrendR95p[parametros[j]][i]=resultado4[i][j]
# R99p
resultado1= PR99p.apply(regresionlineal)
resultado2= VR99p.apply(regresionlineal)
resultado3= OR99p.apply(regresionlineal)
resultado4= IR99p.apply(regresionlineal)
for i in range(0,len(nombre)):
  for j in range(0,len(parametros)):
     PtrendR99p[parametros[j]][i]=resultado1[i][j]
     VtrendR99p[parametros[j]][i]=resultado2[i][j]
     OtrendR99p[parametros[j]][i]=resultado3[i][j]
     ItrendR99p[parametros[j]][i]=resultado4[i][j]
#R95pTot
resultado1= PR95pTot.apply(regresionlineal)
resultado2= VR95pTot.apply(regresionlineal)
resultado3 = OR95pTot.apply(regresionlineal)
resultado4= IR95pTot.apply(regresionlineal)
for i in range(0,len(nombre)):
  for j in range(0,len(parametros)):
     PtrendR95pTot[parametros[j]][i]=resultado1[i][j]
     VtrendR95pTot[parametros[j]][i]=resultado2[i][j]
     OtrendR95pTot[parametros[j]][i]=resultado3[i][j]
     ItrendR95pTot[parametros[j]][i]=resultado4[i][j]
# TEST MANN-KENDALL (Schramm, 2018)
from scipy.stats import norm
def mk_test(x, alpha=0.05):
  n = len(x)
  s = 0
  for k in range(n-1):
    for j in range(k+1, n):
       s += np.sign(x[j] - x[k]) # Calcula S
  unique_x = np.unique(x) # Calcula valores unicos
  g = len(unique_x)
```

```
if n == g:
     var_s = (n*(n-1)*(2*n+5))/18 \# calcula VAR(S) si no hay valores repetidos
  else:
     tp = np.zeros(unique x.shape)
     for i in range(len(unique_x)):
       tp[i] = sum(x == unique_x[i]) #calcula el nº de veces que se repite cada valor "unico"
     var_s = (n^*(n-1)^*(2^*n+5) - np.sum(tp^*(tp-1)^*(2^*tp+5)))/18  # si hay valores repetidos en la
serie
   if s > 0:
     z = (s - 1)/np.sqrt(var_s)
  elif s < 0:
     z = (s + 1)/np.sqrt(var_s)
  else: \# s == 0:
     z = 0
  p = 2*(1-norm.cdf(abs(z))) # calcula el p value
  h = abs(z) > norm.ppf(1-alpha/2)
  if (z < 0) and h:
     trend = 'Decreasing'
  elif (z > 0) and h:
     trend = 'Increasing'
  else:
     trend = 'no trend'
  return trend, p
# Funcion que aplica el Test de MannKendall a una serie de datos y devuelve el resultado
def trends(df,parametro):
  columna = []
  for i in range (0,len(nb)): # Crea 1 columna de nan con el mismo numero de filas que de
estaciones
     columna.append(np.nan)
  df['p value MK'] = columna #crea una columna nueva
  df['Trend MK'] = columna #crea una columna nueva
  for i in range (0,len(nb)):
     datos = parametro[nb[i]][pd.notna(parametro[nb[i]])] #datos a los que se les aplica MK
     x = datos.tolist() #Crea lista de valores
     trend, p = mk_test(x,0.05)
     df['p value MK'][nb[i]] = p
     df['Trend MK'][nb[i]] = trend
  return df
```

tablas = [PtrendPT, VtrendPT, OtrendPT, ItrendPT, Ptrendwd, Vtrendwd, Otrendwd, Itrendwd, PtrendSDII, VtrendSDII, OtrendSDII, ItrendSDII, PtrendCDD, VtrendCDD, OtrendCDD, ItrendCDD, PtrendCWD, VtrendCWD, OtrendCWD, ItrendCWD, PtrendR10, VtrendR10, OtrendR10, ItrendR10, PtrendR20, VtrendR20, OtrendR20, ItrendR20, PtrendR25, VtrendR25, OtrendR25, ItrendR25, PtrendRX1D, VtrendRX1D, OtrendRX1D, ItrendRX1D, PtrendRX5D, VtrendRX5D, OtrendRX5D, OtrendRX5D, VtrendR35D, VtrendR3

```
for i in range (0,len(tablas)):
    a = tablas[i]
    b = indices[i]
    trends(a,b)
```

### #Guarda datos

nombres = ['PtrendPT', 'VtrendPT', 'OtrendPT', 'ItrendPT', 'Ptrendwd', 'Vtrendwd', 'Otrendwd', 'Itrendwd', 'PtrendSDII', 'VtrendSDII', 'OtrendSDII', 'ItrendSDII', 'PtrendCDD', 'VtrendCDD', 'OtrendCDD', 'ItrendCDD', 'PtrendCWD', 'VtrendCWD', 'OtrendCWD', 'ItrendCWD', 'PtrendR10', 'VtrendR10', 'OtrendR10', 'ItrendR10', 'PtrendR20', 'VtrendR20', 'OtrendR20', 'ItrendR20', 'PtrendR25', 'VtrendR25', 'OtrendR25', 'ItrendR25', 'PtrendRX1D', 'VtrendRX1D', 'OtrendRX1D', 'ItrendRX1D', 'PtrendRX5D', 'VtrendRX5D', 'OtrendRX5D', 'ItrendRX5D', 'PtrendR90p', 'VtrendR90p', 'OtrendR90p', 'ItrendR90p', 'PtrendR95p', 'VtrendR95p', 'OtrendR95p', 'ItrendR95p', 'PtrendR99p', 'VtrendR99p', 'OtrendR99p', 'ItrendR99p', 'PtrendR95pTot', 'VtrendR95pTot', 'OtrendR95pTot', 'ItrendR95pTot'] for i in range(0,len(tablas)):

tablas[i].to\_csv(nombres[i]+'.csv',sep=";")