

**DOBLE GRADO DE MATEMÁTICAS Y
ADMINISTRACIÓN Y DIRECCIÓN DE
EMPRESAS**

Trabajo de fin de grado

**ANEXOS: ANÁLISIS DE
DIFERENTES MEDIDAS DE
RIESGO Y APLICACIÓN AL
EUROSTOXX 50**

Autora: Berta de Pablo Brito

Directores: Dr. José Sáez Madrid

Dr. Josep Vives i Santa-Eulàlia

Barcelona, 24 de enero de 2021

Índice

1. ANEXO 1: Tablas correlaciones	3
2. ANEXO 2: Códigos	7

1. ANEXO 1: Tablas correlaciones

A continuación, adjunto las tablas de correlación entre los rankings obtenidos con las distintas medidas de riesgo para 2018, 2017 y 2016.

CORRELACIONES RANKINGS 2018:

	Volatilidad	Pr. Pérd.	Coef. beta	VaR	Rango	Desv. Abs. Me.	Pérd. Esp.	CV	Downs. dev.
Volatilidad	1,00	-0,300	0,530	0,856	0,799	0,933	-0,756	-0,011	0,719
Pr. Pérd.	-0,300	1,00	-0,346	0,064	-0,141	-0,346	-0,061	0,263	0,149
Coef. beta	0,530	-0,346	1,00	0,341	0,293	0,555	-0,243	0,121	0,211
VaR	0,856	0,064	0,341	1,00	0,747	0,775	-0,856	-0,011	0,903
Rango	0,799	-0,141	0,293	0,747	1,00	0,614	-0,784	-0,091	0,712
Desv. Abs. Med	0,933	-0,346	0,555	0,775	0,614	1,00	-0,636	0,010	0,640
Pérd. Esp.	-0,756	-0,061	-0,243	-0,856	-0,784	-0,636	1,00	0,154	-0,967
CV	-0,011	0,263	0,121	-0,011	-0,091	0,010	0,154	1,00	-0,120
Downs. dev.	0,719	0,149	0,211	0,903	0,712	0,640	-0,967	-0,120	1,00

CORRELACIONES RANKINGS 2017:

	Volatilidad	Pr. Pérd.	Coef. beta	VaR	Rango	Desv. Abs. Me.	Pérd. Esp.	CV	Downs. dev.
Volatilidad	1,00	0,098	0,460	0,902	0,830	0,913	-0,799	0,031	0,833
Pr. Pérd.	0,098	1,00	-0,053	-0,200	-0,030	0,090	0,272	-0,292	-0,286
Coef. beta	0,460	-0,053	1,00	0,451	0,240	0,593	-0,376	-0,044	0,463
VaR	0,903	-0,200	0,451	1,00	0,781	0,841	-0,887	-0,001	0,946
Rango	0,830	-0,030	0,240	0,781	1,00	0,618	-0,754	-0,014	0,718
Desv. Abs. Me.	0,913	0,090	0,593	0,841	0,618	1,00	-0,742	0,042	0,799
Pérd. Esp.	-0,799	0,272	-0,376	-0,887	-0,754	-0,742	1,00	0,020	-0,958
CV	0,031	-0,292	-0,044	-0,001	-0,014	0,042	0,020	1,00	-0,046
Downs. dev.	0,833	-0,286	0,463	0,946	0,718	0,799	-0,958	-0,046	1,00

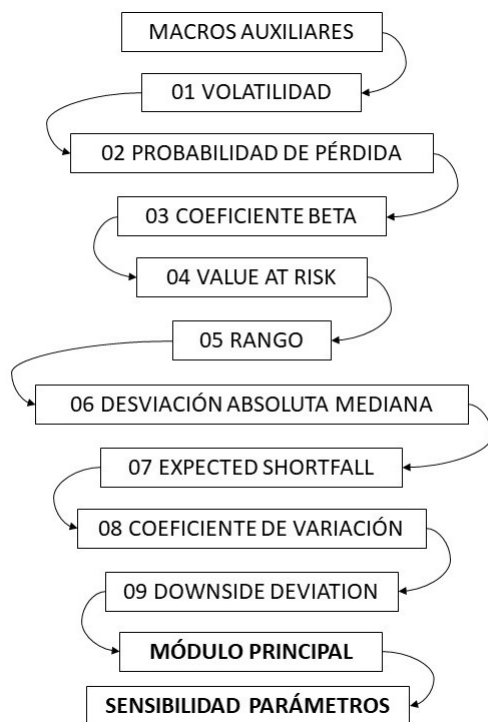
CORRELACIONES RANKINGS 2016:

	Volatilidad	Pr. Pérd.	Coef. beta	VaR	Rango	Desv. Abs. Me.	Pérd. Esp.	CV	Downs. dev.
Volatilidad	1,00	0,024	0,774	0,957	0,866	0,963	-0,801	-0,116	0,862
Pr. Pérd.	0,024	1,00	0,099	-0,194	0,067	0,032	0,156	-0,474	-0,263
Coef. beta	0,774	0,099	1,00	0,752	0,597	0,791	-0,642	-0,294	0,709
VaR	0,957	-0,194	0,752	1,00	0,799	0,931	0,865	-0,090	0,952
Rango	0,866	0,067	0,597	0,799	1,00	0,757	-0,691	-0,070	0,692
Desv. Abs. Me.	0,963	0,032	0,791	0,931	0,757	1,00	-0,773	-0,090	0,886
Pérd. Esp.	-0,801	0,156	-0,642	-0,865	-0,691	-0,773	1,00	0,167	-0,942
CV	-0,116	-0,474	-0,294	-0,090	-0,070	-0,090	0,167	1,00	-0,099
Downs. dev.	0,862	-0,263	0,709	0,952	0,692	0,866	-0,942	-0,099	1,00

2. ANEXO 2: Códigos

Este anexo contiene el código de los programas que he desarrollado a lo largo del trabajo. El primero está desarrollado en SAS y lo he utilizado para sacar las tablas de correlación y analizar la sensibilidad de los parámetros. El segundo está desarrollado en R y lo he utilizado para a verificación de las hipótesis.

Programa 1 (en SAS): El código de este programa contiene distintos módulos que deben ejecutarse en un orden concreto.



Programa 2 (en R): El código de este programa contiene un sólo módulo.

A continuación, adjunto el código de ambos programas, módulo a módulo.


```

/*****
/** MACROS AUXILIARES *****/
/*****

/** 1 MACRO IMPORTAR **/
/*Importar datos del EUROSTOXX 50*/
%MACRO IMPORTAR;
  %DO N=1 %TO &nEMP2;
    %let name = %scan(&empresas, &N);
    %let ruta = "/home/u50152932/TFG/INPUT DATA/Datos históricos &name..csv";
    PROC IMPORT datafile= &ruta
      out= &name
      dbms= csv
      replace;
    RUN;
  %END;
%MEND;

/** 2 MACRO LIMPIAR **/
/* Preparar los datos para el cálculo del riesgo*/

%MACRO LIMPIAR;
  %DO N=1 %TO &nEMP2;
    %let name = %scan(&empresas, &N);

    /*arreglar formatos y calcular rentabilidad*/
    DATA &name;
      set &name;

      Year= substr(Fecha, 7,4);
      Month = substr(Fecha, 4,2);
      Day = substr(Fecha,1,2);
      Close = input (translate(Último, ".", ", ".), comma9.2);

      PriceReturn = Close/lag(Close) - 1;

      keep Year Month Day Close PriceReturn;
    RUN;

    /*añadir variable cantidad de semanas(de cada año)sobre las cuales hay datos */
    PROC SQL;
      create table Tb1 as
      select *, Count(Day) as Nobs
      from &name
      group by Year;
    RUN;

    /*ordenar por fecha*/
    PROC SORT data=Tb1; by Year Month Day; RUN;

    /*Hay empresas que han entrado en bolsa más tarde de enero de 2016.
    Para cada año, solo tenemos en cuenta las empresas que han cotizado todo el año. */
    DATA &name;
      set Tb1;
      if Nobs<52 then delete;
      drop Nobs;
    RUN;

    PROC DELETE data=Tb1; RUN;
  %END;
%MEND;

/**3 MACRO EVOLUCIÓN **/
/* Para cada medida de riesgo, crea una tabla con el valor de la medida para cada año (columnas)
y para cada empresa (filas)*/
%MACRO EVOLUCION;
  %DO F=1 %TO &nFunc;

```

```

%let risk_func = %scan(&funciones, &F);
%let name1 = %scan(&empresas, 1);

DATA Evolucion_&risk_func;
  set &risk_func._&name1;
RUN;

%DO N=2 %TO &nEmp;
  %let name = %scan(&empresas, &N);
  DATA Evolucion_&risk_func;
    set Evolucion_&risk_func &risk_func._&name ;
  RUN;
%END;

%END;
%MEND;

/*4 MACRO RANKING*/
/* Para cada medida de riesgo, partiendo del output de la macro EVOLUCIÓN,
crea una tabla con la posición en el ranking de la medida para cada año (columnas)
y para cada empresa (filas)*/
%MACRO RANKING;
  %DO F=1 %TO &nFunc;
    %let risk_func = %scan(&funciones, &F);

    PROC RANK data = Evolucion_&risk_func out=Rank_&risk_func ties=mean;
      var Y2016 Y2017 Y2018 Y2019;
      ranks R2016 R2017 R2018 R2019;
    RUN;

    DATA Rank_&risk_func;
      set Rank_&risk_func;
      drop Y2016 Y2017 Y2018 Y2019;
    RUN;

  %END;
%MEND;

/*5 MACRO RANKING_YEAR*/
/*Partiendo del output de la macro anterior, crea tantas tablas como años.
Cada fila es una empresa y cada columna una medida, y contiene las posiciones en los rankings
anuales. */
%MACRO RANKING_YEAR;
  %DO Y=2016 %TO 2019;
    %let risk_func = %scan(&funciones, 1);
    DATA Rank_&Y; set RANK_&risk_func; nCompany=_N_; &risk_func=R&Y; keep nCompany &risk_func; RUN;

    %DO F=2 %TO &nFunc;
      %let risk_func = %scan(&funciones, &F);
      DATA Add; set RANK_&risk_func; nCompany=_N_; &risk_func=R&Y; keep nCompany &risk_func; RUN;
      PROC SQL;
        create table Rank_&Y as select *
          from Rank_&Y, Add
         where Rank_&Y..nCompany = Add.nCompany;
      QUIT;
    %END;
  %END;
%MEND;

/*6 MACRO CORRELATION*/
%MACRO CORRELATION;
  ods select Cov PearsonCorr;
  %DO Y=2016 %TO 2019;
    proc corr data=Rank_&Y noprob outp=Corr&Y pearson nomiss;
      var &funciones;
    run;
  %END;

```

```

/** VOLATILIDAD **/
/*OUTPUT: Una tabla para cada empresa + una para el EUROSTOXX en general,
con el valor de la volatilidad para cada año (5 columnas, 1 fila):
Y2016 | Y2017 | Y2018 | Y2019 | EMPRESA*/

```

```

%MACRO VOLATILIDAD;
  %DO N=1 %TO &nEMP2;
    %let name = %scan(&empresas, &N);

    PROC SQL;
      create table Tabla1 as
      select Year,
             Mean(PriceReturn) as MeanPriceReturn
             from &name
      where Year in ("2016", "2017", "2018", "2019")
      group by Year;
    QUIT;

    PROC SQL;
      create table Tabla2 as
      select &name..Year, PriceReturn, MeanPriceReturn
             from Tabla1, &name
      where Tabla1.Year = &name..Year;
    QUIT;

    DATA Tabla3;
      set Tabla2;
      Sumando = (PriceReturn - MeanPriceReturn)**2;
    RUN;

    PROC SQL;
      create table Tabla4 as
      select Year, Sum(Sumando) as Suma, Count(PriceReturn) as Nweeks from Tabla3
      group by Year;
    QUIT;

    DATA Volatilidad_&name;
      set Tabla4;
      Volatilidad_&name = sqrt(Suma/Nweeks);
      keep Year Volatilidad_&name;
    RUN;

    PROC TRANSPOSE data=Volatilidad_&name out=Volatilidad_&name prefix=Y;
      id Year;
    RUN;

    DATA Volatilidad_&name;
      set Volatilidad_&name;
      length EMPRESA $50;
      EMPRESA = _NAME_;
      drop _NAME_;
    RUN;

    PROC DELETE data = Tabla1 Tabla2 Tabla3 Tabla4;
    RUN;
  %END;
%MEND;

```

```

/** PROBABILIDAD DE PÉRDIDA **/
/*OUTPUT: Una tabla para cada empresa + una para el EUROSTOXX en general,
con el valor de la probabilidad de pérdida para cada año (5 columnas, 1 fila):
Y2016 | Y2017 | Y2018 | Y2019 | EMPRESA*/
.....
%MACRO PROBABILIDAD_PERDIDA;
  %DO N=1 %TO &nEmp;
    %let name = %scan(&empresas, &N);

    PROC TRANSPOSE data= Volatilidad_&name out=Tabla1 name=Year;
    RUN;

    PROC SQL;
      create table Tabla2 as
      select Year,
             Mean(PriceReturn) as MeanPriceReturn
      from &name
      where Year in ("2016","2017","2018","2019")
      group by Year;
    QUIT;

    DATA Tabla3;
      set Tabla2;
      if Year="2016" then Year="Y2016";
      if Year="2017" then Year="Y2017";
      if Year="2018" then Year="Y2018";
      if Year="2019" then Year="Y2019";
    RUN;

    PROC SQL;
      create table Tabla4 as
      select Tabla1.Year, Tabla1.COL1 as Volatilidad, Tabla3.MeanPriceReturn from Tabla1, Tabla3
      where Tabla1.Year=Tabla3.Year;
    QUIT;

    DATA Tabla5;
      set Tabla4;
      Est = (&R-MeanPriceReturn)/Volatilidad;
      Probabilidad_Perdida_&name = PDF("NORMAL", Est);
      keep Year Probabilidad_Perdida_&name;
    RUN;

    PROC TRANSPOSE data=Tabla5 out=Probabilidad_Perdida_&name;
      id Year;
    RUN;

    DATA Probabilidad_Perdida_&name;
      set Probabilidad_Perdida_&name;
      length EMPRESA $50;
      EMPRESA = _NAME_;
      drop _NAME_;
    RUN;

    PROC DELETE data=Tabla1 Tabla2 Tabla3 Tabla4 Tabla5; RUN;
  %END;
%MEND;

```

```

/** COEFICIENTE BETA **/
/*OUTPUT: Una tabla para cada empresa + una para el EUROSTOXX en general,
con el valor de la coeficiente beta para cada año (5 columnas, 1 fila):
Y2016 | Y2017 | Y2018 | Y2019 | EMPRESA*/

```

```

%MACRO COEFICIENTE_BETA;
  %DO N=1 %TO &nEMP;
    %let name = %scan(&empresas, &N);

    PROC SQL;
      create table Tabla1 as
      select EUROSTOXX50.Year, EUROSTOXX50.Month, EUROSTOXX50.Day,
             &name..PriceReturn as PR_&name,
             EUROSTOXX50.PriceReturn as PR_EUROSTOXX50
      from &name, EUROSTOXX50
      where &name..Year=EUROSTOXX50.Year
            and &name..Month=EUROSTOXX50.Month
            and &name..Day=EUROSTOXX50.Day
            and EUROSTOXX50.Year ^= "2020" ;

    QUIT;

    %DO Y=2016 %TO 2019;

      DATA Tabla1_&Y;
        set Tabla1;
        Year="Y&Y";
        where Year="&Y";
      RUN;

      PROC REG DATA = Tabla1_&Y outest=Reg_&name._&Y plots=none;
        MODEL PR_&name = PR_EUROSTOXX50;
      RUN;

      DATA Reg_&name._&Y;
        set Reg_&name._&Y;
        Year="Y&Y";
        COEFICIENTE_BETA=abs(PR_EUROSTOXX50);
        keep Year COEFICIENTE_BETA;
      RUN;

    %END;

    DATA Reg;
      set Reg_&name._2016 Reg_&name._2017 Reg_&name._2018 Reg_&name._2019;
    RUN;

    PROC TRANSPOSE data=Reg out=Coeficiente_beta_&name;
      id Year;
    RUN;

    DATA Coeficiente_beta_&name;
      set Coeficiente_beta_&name;
      length EMPRESA $50;
      EMPRESA = "Coeficiente_beta_&name";

```

```
        drop _NAME_;  
    RUN;  
  
    %DO Y=2016 %TO 2019;  
        PROC DELETE data=Tabla1_&Y Reg_&name._&Y;  
        RUN;  
    %END;  
  
    PROC DELETE data=Tabla1 Reg;  
    RUN;  
  
%END;  
%MEND;
```

```

/** RANGO **/
/*OUTPUT: Una tabla para cada empresa + una para el EUROSTOXX en general,
con el valor del rango para cada año (5 columnas, 1 fila):
Y2016 | Y2017 | Y2018 | Y2019 | EMPRESA*/

```

```

%MACRO RANGO;
  %DO N=1 %TO &nEMP;
    %let name = %scan(&empresas, &N);

    PROC SQL;
      create table Tabla1 as
      select  Year,
              max(PriceReturn) as maxPR,
              min(PriceReturn) as minPR
      from &name
      where Year ^= "2020"
      group by Year;
    QUIT;

    DATA Tabla2;
      set Tabla1;
      Year=cat("Y",Year,"");
      Rango=maxPR-minPR;
      keep Year Rango;
    RUN;

    PROC TRANSPOSE data=Tabla2 out=Rango_&name;
      id Year;
    RUN;

    DATA Rango_&name;
      set Rango_&name;
      length EMPRESA $50;
      EMPRESA = "Rango_&name";
      drop _NAME_;
    RUN;

    PROC DELETE data=Tabla1 Tabla2; RUN;

    RUN;
  %END;
%MEND;

```

```

/** VALUE AT RISK */
/*OUTPUT: Una tabla para cada empresa + una para el EUROSTOXX en general,
con el valor del VaR para cada año (5 columnas, 1 fila):
Y2016 | Y2017 | Y2018 | Y2019 | EMPRESA*/

```

```

%MACRO VALUE_AT_RISK;
  %DO N=1 %TO &nEMP;
    %let name = %scan(&empresas, &N);

    PROC SQL;
      create table Tabla1 as
      select Year, Mean(PriceReturn) as MeanPriceReturn
      from &name
      where Year ^= "2020"
      group by Year;
    QUIT;

    DATA Tabla1;
      set Tabla1;
      Year=cat("Y",Year,"");
    RUN;

    PROC TRANSPOSE data= Volatilidad_&name out=Tabla2 name=Year;
    RUN;

    PROC SQL;
      create table Tabla3 as
      select Tabla1.Year,
             Tabla1.MeanPriceReturn,
             Tabla2.COL1 as Volatilidad
      from Tabla1, Tabla2
      where Tabla1.Year=Tabla2.Year;
    QUIT;

    DATA Value_at_Risk;
      set Tabla3;
      Value_at_Risk= abs(min(0,MeanPriceReturn- &k * Volatilidad));
      keep Year Value_at_Risk;
    RUN;

    PROC TRANSPOSE data=Value_at_Risk out=Value_at_Risk_&name;
      id Year;
    RUN;

    DATA Value_at_Risk_&name;
      set Value_at_Risk_&name;
      length EMPRESA $50;
      EMPRESA = "Value_at_Risk_&name";
      drop _NAME_;
    RUN;

```



```
PROC DELETE data=Tabla1 Tabla2 Tabla3 Value_at_Risk;  
RUN;  
%END;  
%MEND;
```

```

/** DESVIACIÓN ABSOLUTA MEDIANA **/
/*OUTPUT: Una tabla para cada empresa + una para el EUROSTOXX en general,
con el valor de la desviación absoluta mediana para cada año (5 columnas, 1 fila):
Y2016 | Y2017 | Y2018 | Y2019 | EMPRESA*/

```

```

%MACRO DESV_ABS_MEDIANA;
  %DO N=1 %TO &nEMP;
    %let name = %scan(&empresas, &N);

    PROC SQL;
      create table Tabla1 as
      select Year,
             Median(PriceReturn) as MedianPriceReturn
      from &name
      where Year in ("2016", "2017", "2018", "2019")
      group by Year;
    QUIT;

    PROC SQL;
      create table Tabla2 as
      select &name..Year, PriceReturn, MedianPriceReturn
      from Tabla1, &name
      where Tabla1.Year = &name..Year;
    QUIT;

    DATA Tabla3;
      set Tabla2;
      Sumando = ABS(PriceReturn - MedianPriceReturn);
    RUN;

    PROC SQL;
      create table Tabla4 as
      select Year, Sum(Sumando) as Suma, Count(PriceReturn) as Nweeks from Tabla3
      group by Year;
    QUIT;

    DATA Desv_abs_mediana_&name;
      set Tabla4;
      Desv_abs_mediana_&name = Suma/Nweeks;
      keep Year Desv_abs_mediana_&name;
    RUN;

    PROC TRANSPOSE data=Desv_abs_mediana_&name out=Desv_abs_mediana_&name prefix=Y;
      id Year;
    RUN;

    DATA Desv_abs_mediana_&name;
      set Desv_abs_mediana_&name;
      length EMPRESA $50;
      EMPRESA = _NAME_;
      drop _NAME_;
    RUN;

    PROC DELETE data = Tabla1 Tabla2 Tabla3 Tabla4;
    RUN;
  %END;
%MEND;

```

```

/** PÉRDIDA ESPERADA **/
/*OUTPUT: Una tabla para cada empresa + una para el EUROSTOXX en general,
con el valor de la pérdida esperada para cada año (5 columnas, 1 fila):
Y2016 | Y2017 | Y2018 | Y2019 | EMPRESA*/

```

```

%MACRO EXPECTED_SHORTFALL;
  %DO N=1 %TO &nEMP;
    %let name = %scan(&empresas, &N);

    PROC SQL;
      create table Tabla0 as select
        Year, PriceReturn from &name
      group by Year
      order by Year, PriceReturn;
    QUIT;

    DATA Tabla0;
      set Tabla0;
      by Year;
      if first.Year then Count=0;
      Count+1;
    RUN;

    DATA Tabla0;
      set Tabla0;
      where Count < &lim;
    RUN;

    PROC SQL;
      create table Expected_shortfall_&name as select
        Year, Mean(PriceReturn) as Expected_Shortfall from Tabla0
      group by Year;
    QUIT;

    PROC TRANSPOSE data= Expected_shortfall_&name out= Expected_shortfall_&name prefix=Y;
      id Year;
    RUN;

    DATA Expected_Shortfall_&name;
      set Expected_Shortfall_&name;
      length EMPRESA $50;
      EMPRESA = "Expected_Shortfall_&name";
      drop _NAME_;
    RUN;

    PROC DELETE data=Tabla0; RUN;

  %END;
%MEND;

```

```

/** COEFICIENTE DE VARIACIÓN **/
/*OUTPUT: Una tabla para cada empresa + una para el EUROSTOXX en general,
con el valor del coeficiente de variación para cada año (5 columnas, 1 fila):
Y2016 | Y2017 | Y2018 | Y2019 | EMPRESA*/

```

```

%MACRO COEFICIENTE_VARIACION;
  %DO N=1 %TO &nEMP;
    %let name = %scan(&empresas, &N);

    PROC SQL;
      create table Tabla1 as
      select Year,
             Mean(PriceReturn) as MeanPriceReturn
             from &name
      where Year in ("2016", "2017", "2018", "2019")
      group by Year;
    QUIT;

    PROC SQL;
      create table Tabla2 as
      select &name..Year, PriceReturn, MeanPriceReturn
             from Tabla1, &name
      where Tabla1.Year = &name..Year;
    QUIT;

    DATA Tabla3;
      set Tabla2;
      Sumando = (PriceReturn - MeanPriceReturn)**2;
    RUN;

    PROC SQL;
      create table Tabla4 as
      select Year, Sum(Sumando) as Suma, Count(PriceReturn) as Nweeks, Max(MeanPriceReturn) as ExpectedValue
      from Tabla3
      group by Year;
    QUIT;

    DATA Coeficiente_Variacion_&name;
      set Tabla4 ;
      Volatilidad = sqrt(Suma/Nweeks);
      Coeficiente_Variacion_&name=Volatilidad/ExpectedValue;
      keep Year Coeficiente_Variacion_&name;
    RUN;

    PROC TRANSPOSE data=Coeficiente_Variacion_&name out=Coeficiente_Variacion_&name prefix=Y;
      id Year;
    RUN;

    DATA Coeficiente_Variacion_&name;
      set Coeficiente_Variacion_&name;
      length EMPRESA $50;
      EMPRESA = _NAME_;
      drop _NAME_;
    RUN;

    PROC DELETE data = Tabla1 Tabla2 Tabla3 Tabla4;
    RUN;

  %END;
%MEND;

```

```

/** DOWNSIDE_DEVIATION */
/*OUTPUT: Una tabla para cada empresa + una para el EUROSTOXX en general,
con el valor de la downside deviation para cada año (5 columnas, 1 fila):
Y2016 | Y2017 | Y2018 | Y2019 | EMPRESA*/

```

```

%MACRO DOWNSIDE_DEVIATION;

```

```

%DO N=1 %TO &nEMP;
  %let name = %scan(&empresas, &N);

```

```

  DATA Tabla0;
    set &name;
    if PriceReturn < &R then D = PriceReturn- &R;
    else D=0;
    D2=D*D;
  RUN;

```

```

  PROC SQL;
    create table Tabla1 as select
    Year, Sum(D2) as Suma, count(Year) as Nweeks from Tabla0
    group by Year;
  QUIT;

```

```

  DATA Downside_deviation_&name;
    set Tabla1;
    Downside_deviation_&name = sqrt(Suma/Nweeks);
    keep Year Downside_deviation_&name;
  RUN;

```

```

  PROC TRANSPOSE data=Downside_deviation_&name out=Downside_deviation_&name prefix=Y;
    id Year;
  RUN;

```

```

  DATA Downside_deviation_&name;
    set Downside_deviation_&name;
    length EMPRESA $50;
    EMPRESA = _NAME_;
    drop _NAME_;
  RUN;

```

```

  PROC DELETE data=Tabla0 Tabla1; RUN;

```

```

%END;

```

```

%MEND;

```

```

/*****/
/** MÓDULO PRINCIPAL ***/
/*****/

/** PASO 0: Lectura de macrovariables**/
* Lista de empresas del eurostoxx 50;
%let empresas = ABI AD ADGn ADYEN AIR AIRP ALVG AMA ASML AXAF BASFn BAYGn BMWG
  BNPP CRH DAIGn DANO DB1Gn DPWgN DTEGn ENEI ENGIE ENI ESLX IBE INGA ISP ITX
  KNEBV LINI LVMH MUVGn NOKIA OREP PERP PHG PRTP PRX SAF SAN SAPG SASY SCHN SGEF
  SIEGn TOTF UNA VIV VAn VOWG_p EUROSTOXX50;

%let nEMP = 50;*número de empresas;
%let nEMP2 = 51;*número de empresas + eurostoxx en general;

* Lista medidas de riesgo;
%let funciones = Volatilidad Probabilidad_Perdida Coeficiente_Beta
  Value_at_Risk Rango Desv_abs_mediana Expected_Shortfall
  Coeficiente_Variacion Downside_deviation;

%let nFunc = 9;*número de medidas de riesgo;

/** PASO 1: Importación y limpieza de datos del EUROSTOXX **/
%IMPORTAR;
%LIMPIAR;

/** PASO 2: Cálculo del riesgo para cada medida y año **/

%VOLATILIDAD;

%let R = 0;*Parámetro: rentabilidad deseada = 0;
%PROBABILIDAD_PERDIDA;

%COEFICIENTE_BETA;

%let k=1.645;*Parámetro equivalente a nivel de confianza 95%;
%VALUE_AT_RISK;

%RANGO;

%DESV_ABS_MEDIANA;

%let lim = 6; *Parámetro equivalente a q=10%;
%EXPECTED_SHORTFALL;

%COEFICIENTE_VARIACION;

%DOWNSIDE_DEVIATION;

/** PASO 3: Para cada medida y año, crea el ranking de las empresas del EUROSTOXX según
el valor de la medida de riesgo **/
%EVOLUCION;
%RANKING;
%RANKING_YEAR;

/** PASO 4: Calcular las correlaciones **/
%CORRELATION;

```

```

/*****
** SENSIBILIDAD A LOS PARÁMETROS **
*****/

/** 1 - PROBABILIDAD DE PÉRDIDA **/
/*¿Cómo un cambio en la rentabilidad deseada afecta al ranking de la probabilidad de pérdida?*/
%let funciones = Probabilidad_Perdida;
%let nFunc = 1;

-----
%MACRO SENS_PROB_PERD;
    %PROBABILIDAD_PERDIDA;
    %EVOLUCION;
    %RANKING;

    DATA SENS_PR_PERDIDA_&Valor ;
        set RANK_PROBABILIDAD_PERDIDA (rename=(R2019=R_&Valor));
        keep EMPRESA R_&Valor;
    RUN;
%MEND;

/*Generar el ranking para 5 valores distintos de la rentabilidad deseada*/
%let Valor=1;
%let R=0;
%SENS_PROB_PERD;

%let Valor=2;
%let R=0.025;
%SENS_PROB_PERD;

%let Valor=3;
%let R=0.05;
%SENS_PROB_PERD;

%let Valor=4;
%let R=0.075;
%SENS_PROB_PERD;

%let Valor=5;
%let R=0.1;
%SENS_PROB_PERD;

/*Calcular la correlación entre rankings*/
PROC SQL; create table SENS_PR_PERDIDA as select * from
SENS_PR_PERDIDA_1, SENS_PR_PERDIDA_2, SENS_PR_PERDIDA_3, SENS_PR_PERDIDA_4, SENS_PR_PERDIDA_5
where SENS_PR_PERDIDA_1.EMPRESA=SENS_PR_PERDIDA_2.EMPRESA
and SENS_PR_PERDIDA_1.EMPRESA=SENS_PR_PERDIDA_3.EMPRESA
and SENS_PR_PERDIDA_1.EMPRESA=SENS_PR_PERDIDA_4.EMPRESA
and SENS_PR_PERDIDA_1.EMPRESA=SENS_PR_PERDIDA_5.EMPRESA
;
QUIT;

-----
proc corr data=SENS_PR_PERDIDA pearson
    plots=matrix(histogram);
var R_1 R_2 R_3 R_4 R_5;
run;

/** 2 VALUE AT RISK **/
/*¿Cómo un cambio en el nivel de confianza afecta al ranking de la probabilidad de pérdida?*/
%let funciones = Value_at_Risk;
%let nFunc = 1;

-----
%MACRO SENS_VAR;
    %VALUE_AT_RISK;

```

```

%EVOLUCION;
%RANKING;

DATA SENS_VAR_&Valor ;
    set RANK_VALUE_AT_RISK (rename=(R2019=K_&Valor));
    keep EMPRESA K_&Valor;
RUN;
%MEND;

/*Generar el ranking para 5 valores distintos del nivel de confianza*/
%let Valor=1;
%let k=0.842; *nivel confianza 80%;
%SENS_VAR;

%let Valor=2;
%let k=1.036; *nivel confianza 85%;
%SENS_VAR;

%let Valor=3;
%let k=1.282; *nivel confianza 90%;
%SENS_VAR;

%let Valor=4;
%let k=1.645; *nivel confianza 95%;
%SENS_VAR;

%let Valor=5;
%let k=2.326; *nivel confianza 99%;
%SENS_VAR;

/*Calcular la correlación entre rankings*/
PROC SQL;
    create table SENS_VAR as
    select
        SENS_VAR_1.EMPRESA, K_1, K_2, K_3, K_4, K_5
    from SENS_VAR_1, SENS_VAR_2, SENS_VAR_3, SENS_VAR_4, SENS_VAR_5
    where SENS_VAR_1.EMPRESA = SENS_VAR_2.EMPRESA
        and SENS_VAR_1.EMPRESA = SENS_VAR_3.EMPRESA
        and SENS_VAR_1.EMPRESA = SENS_VAR_4.EMPRESA
        and SENS_VAR_1.EMPRESA = SENS_VAR_5.EMPRESA;
QUIT;

proc corr data=SENS_VAR pearson
    plots=matrix(histogram);
var K_1 K_2 K_3 K_4 K_5;
run;

/** 3 PÉRDIDA ESPERADA**/
/*¿Cómo un cambio en el parámetro N' afecta al ranking de la pérdida esperada?*/
%let funciones = Expected_Shortfall;
%let nFunc = 1;

%MACRO SENS_ES;
    %EXPECTED_SHORTFALL;
    %EVOLUCION;
    %RANKING;

    DATA SENS_ES_&Valor ;
        set RANK_EXPECTED_SHORTFALL (rename=(R2019=N_&Valor));
        keep EMPRESA N_&Valor;
    RUN;
%MEND;

/*Generar el ranking para 5 valores distintos de N'*/

```



```

%let Valor=1;
%let NumEmp=1;
%SENS_ES;

%let Valor=2;
%let NumEmp=3;
%SENS_ES;

%let Valor=3;
%let NumEmp=5;
%SENS_ES;

%let Valor=4;
%let NumEmp=8;
%SENS_ES;

%let Valor=5;
%let NumEmp=10;
%SENS_ES;

/*Calcular la correlación entre rankings*/
PROC SQL;
  create table SENS_ES as
  select
    SENS_ES_1.EMPRESA, N_1, N_2, N_3, N_4, N_5
  from SENS_ES_1, SENS_ES_2, SENS_ES_3, SENS_ES_4, SENS_ES_5
  where SENS_ES_1.EMPRESA = SENS_ES_2.EMPRESA
    and SENS_ES_1.EMPRESA = SENS_ES_3.EMPRESA
    and SENS_ES_1.EMPRESA = SENS_ES_4.EMPRESA
    and SENS_ES_1.EMPRESA = SENS_ES_5.EMPRESA;
QUIT;

proc corr data=SENS_ES pearson
  plots=matrix(histogram);
var N_1 N_2 N_3 N_4 N_5;
run;

/** 4 DOWNSIDE DEVIATION**/
/*¿Cómo un cambio en la rentabilidad deseada afecta al ranking de la downside deviation?*/
%let funciones = Downside_deviation;
%let nFunc = 1;

%MACRO SENS_DD;
  %DOWNSIDE_DEVIATION;
  %EVOLUCION;
  %RANKING;

  DATA SENS_DD_&Valor ;
    set RANK_DOWNSIDE_DEVIATION (rename=(R2019=R_&Valor));
    keep EMPRESA R_&Valor;
  RUN;
%MEND;

/*Generar el ranking para 5 valores distintos de la rentabilidad deseada*/
%let Valor=1;
%let R=0;
%SENS_DD;

%let Valor=2;
%let R=0.025;
%SENS_DD;

%let Valor=3;
%let R=0.05;

```

```

%SENS_DD;

%let Valor=4;
%let R=0.075;
%SENS_DD;

%let Valor=5;
%let R=0.1;
%SENS_DD;

/*Calcular la correlación entre rankings*/
PROC SQL;
  create table SENS_DD as
  select
    SENS_DD_1.EMPRESA, R_1, R_2, R_3, R_4, R_5
  from SENS_DD_1, SENS_DD_2, SENS_DD_3, SENS_DD_4, SENS_DD_5
  where SENS_DD_1.EMPRESA = SENS_DD_2.EMPRESA
    and SENS_DD_1.EMPRESA = SENS_DD_3.EMPRESA
    and SENS_DD_1.EMPRESA = SENS_DD_4.EMPRESA
    and SENS_DD_1.EMPRESA = SENS_DD_5.EMPRESA;
QUIT;

proc corr data=SENS_DD pearson
  plots=matrix(histogram);
  var R_1 R_2 R_3 R_4 R_5;
run;

```

```

#####
### VERIFICACIÓN DE LAS HIPÓTESIS ###
#####

library(mondate)
library(readxl)
library(sqldf)
library(nortest)

#Importar y limpiar datos, y calcular rentabilidad semanal para el EUROSTOXX
Companies <- read_excel("E:/TFG/EUROSTOXX50 2015-2019/Companies_list.xlsx")[,3]

EUROSTOXX50<-as.data.frame(read.csv("E:/TFG/EUROSTOXX50 2015-2019/Datos históricos Euro Stoxx 50.csv"))[,1:2]
colnames(EUROSTOXX50)<-c("Date", "close")

EUROSTOXX50$close <- sub(".", "", EUROSTOXX50$close, fixed=TRUE)
EUROSTOXX50$close <- as.numeric(sub(",", ".", EUROSTOXX50$close, fixed=TRUE))

EUROSTOXX50$PR <- NA
for(i in 2:nrow(EUROSTOXX50)){
  PR <- as.numeric(EUROSTOXX50$close[i-1] / EUROSTOXX50$close[i] -1)
  EUROSTOXX50$PR[i] <- PR
}

EUROSTOXX50$Year<-as.numeric(substr(EUROSTOXX50$Date, 7, 10))
EUROSTOXX50<-sqldf('SELECT PR FROM EUROSTOXX50 WHERE Year=2019 ')

#Inicializar tabla hipótesis normalidad
pvalue<-data.frame(matrix(NA, nrow = 50, ncol = 2))
colnames(pvalue)<- c("p value", "Normal distribution 0.05")
alpha=0.05

#Inicializar tabla hipótesis correlación con el mercado
Correl<-data.frame(matrix(NA, nrow = 50, ncol = 2))
colnames(Correl)<- c("Coeficiente R^2", "Conclusión")

#verificación de las hipótesis para cada empresa
for (n in 1:50){

  #Importar y limpiar datos, y calcular rentabilidad semanal para la empresa
  CompName <- Companies[n,1]
  ExcelName <- paste("Datos históricos ",CompName, ".csv", sep="")
  COMP<-as.data.frame(read.csv(paste("E:/TFG/EUROSTOXX50 2015-2019/", ExcelName, sep="")))[,1:2]
  colnames(COMP)<-c("Date", "close")

  COMP$close <- sub(".", "", COMP$close, fixed=TRUE)
  COMP$close <- as.numeric(sub(",", ".", COMP$close, fixed=TRUE))

  COMP$PR <- NA
  for(i in 2:nrow(COMP)){
    PR <- as.numeric(COMP$close[i-1] / COMP$close[i] -1)
    COMP$PR[i] <- PR
  }

  COMP$Year<-as.numeric(substr(COMP$Date, 7, 10))
  COMP<-sqldf('SELECT PR FROM COMP WHERE Year=2019 ')

  #TEST DE NORMALIDAD
  pvalue$p value[n] <-lillie.test(COMP$PR)$p.value

  if(pvalue$p value[n] < alpha){
    pvalue$Normal distribution 0.05[n] = "RECHAZO H0"
  }else{
    pvalue$Normal distribution 0.05[n] = "ACEPTO H0"
  }

  #CÁLCULO COEFICIENTE DE DETERMINACIÓN
  Correl$Coeficiente R^2[n] <- cor(COMP$PR, EUROSTOXX50$PR)**2
  if(Correl$Coeficiente R^2[n] <0.8){
    Correl$conclusión[n]<- "NO DEPENDENCIA LINEAL"
  }else{
    Correl$conclusión[n] <- "DEPENDENCIA LINEAL"
  }
}
}

```