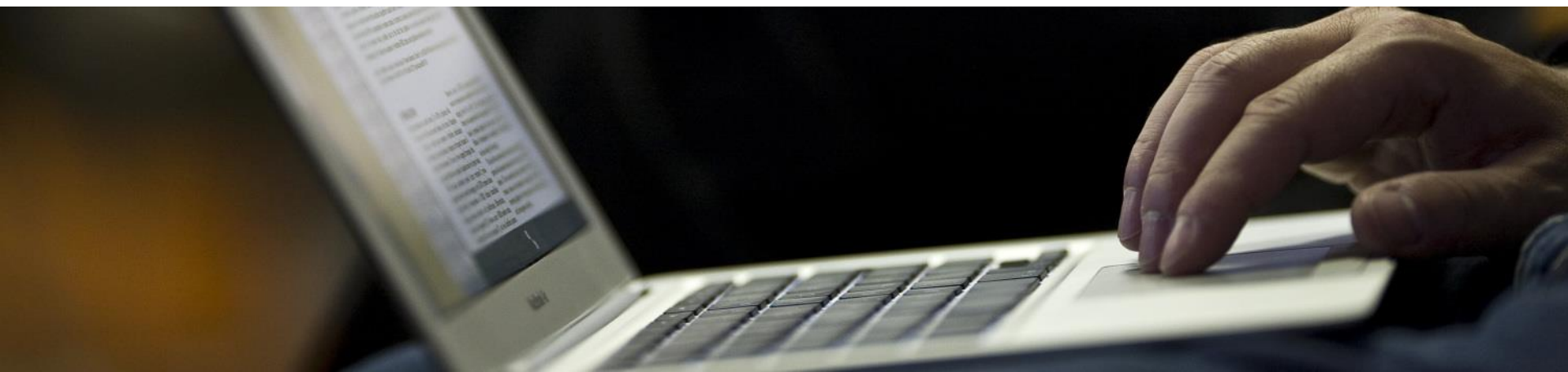

Q-Q plots as an instrument for comparing distributions (SP02)

Holger Langkabel
PhUSE 2016



Introduction

Probability Plots in Theory

Probability Plots in Practice

Q-Q Plot Empirical vs. Empirical

Conclusion

Introduction

Probability Plots in Theory

Probability Plots in Practice

Q-Q Plot Empirical vs. Empirical

Conclusion

Relationship between Theoretical and Empirical Quantiles

- Collect a sample $x_1, x_2, \dots, x_n$,
- Sort the observations in ascending order $x_{(1)}, x_{(2)}, \dots, x_{(n)}$, and
- Define a cumulative density function $F(x)$.
- Then the theoretical probability will approximately equal its empirical estimate:

$$F(x_{(i)}) \approx \frac{i}{n}.$$

- Re-arrangement yields:

$$x_{(i)} \approx F^{-1}\left(\frac{i}{n}\right).$$

- I.e. theoretical and empirical quantiles will be approximately the same, too.

Instructions for Q-Q Plots

- Plot the sorted observations against the theoretically derived quantiles.
- If the assumed distribution is correct, the marks will randomly scatter around a line with slope 1 and intercept 0.

Introduction

Probability Plots in Theory

Probability Plots in Practice

Q-Q Plot Empirical vs. Empirical

Conclusion

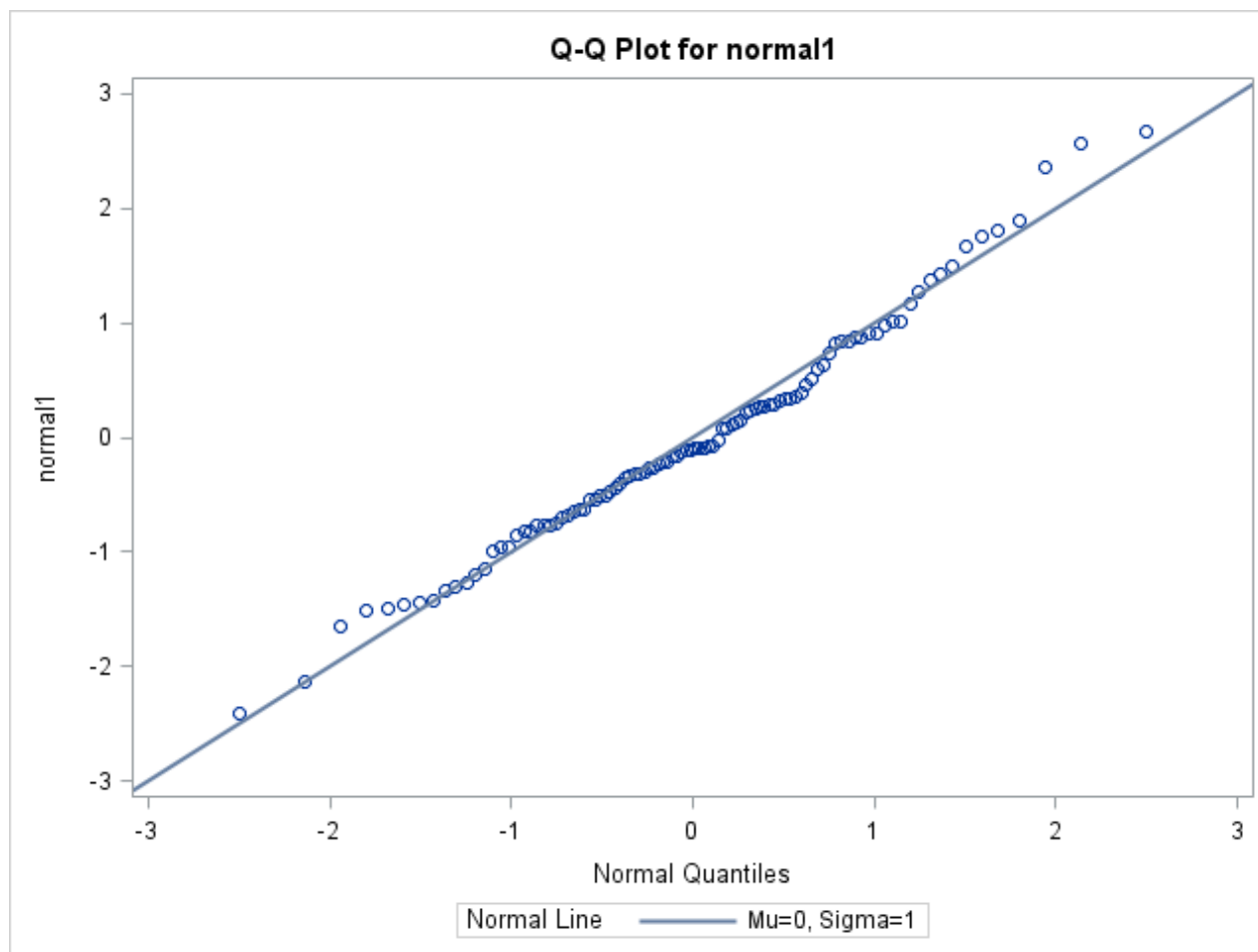
Example Data

```
data normal;  
    call streaminit(1234);  
    do i=1 to 100;  
        normal1 = rand('normal',0,1);  
        normal2 = rand('normal',10,4);  
        exp = rand('weibull',1,0.5);  
        output;  
    end;  
run;
```

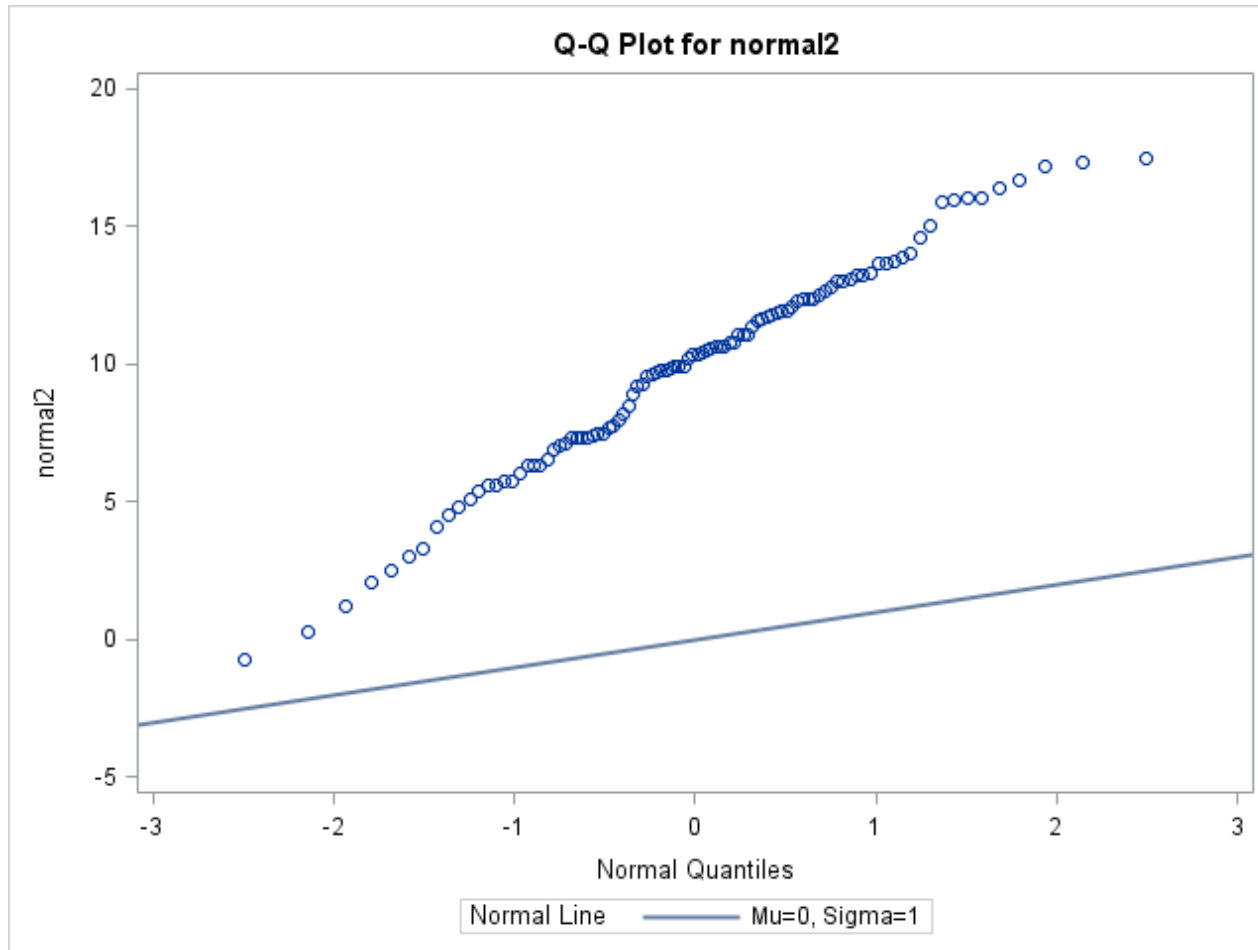
Q-Q Plots of the PROC UNIVARIATE

```
proc univariate data=normal noprint;  
    qqplot normal1 normal2 exp  
        / normal(mu=0 sigma=1) ;  
  
run;
```

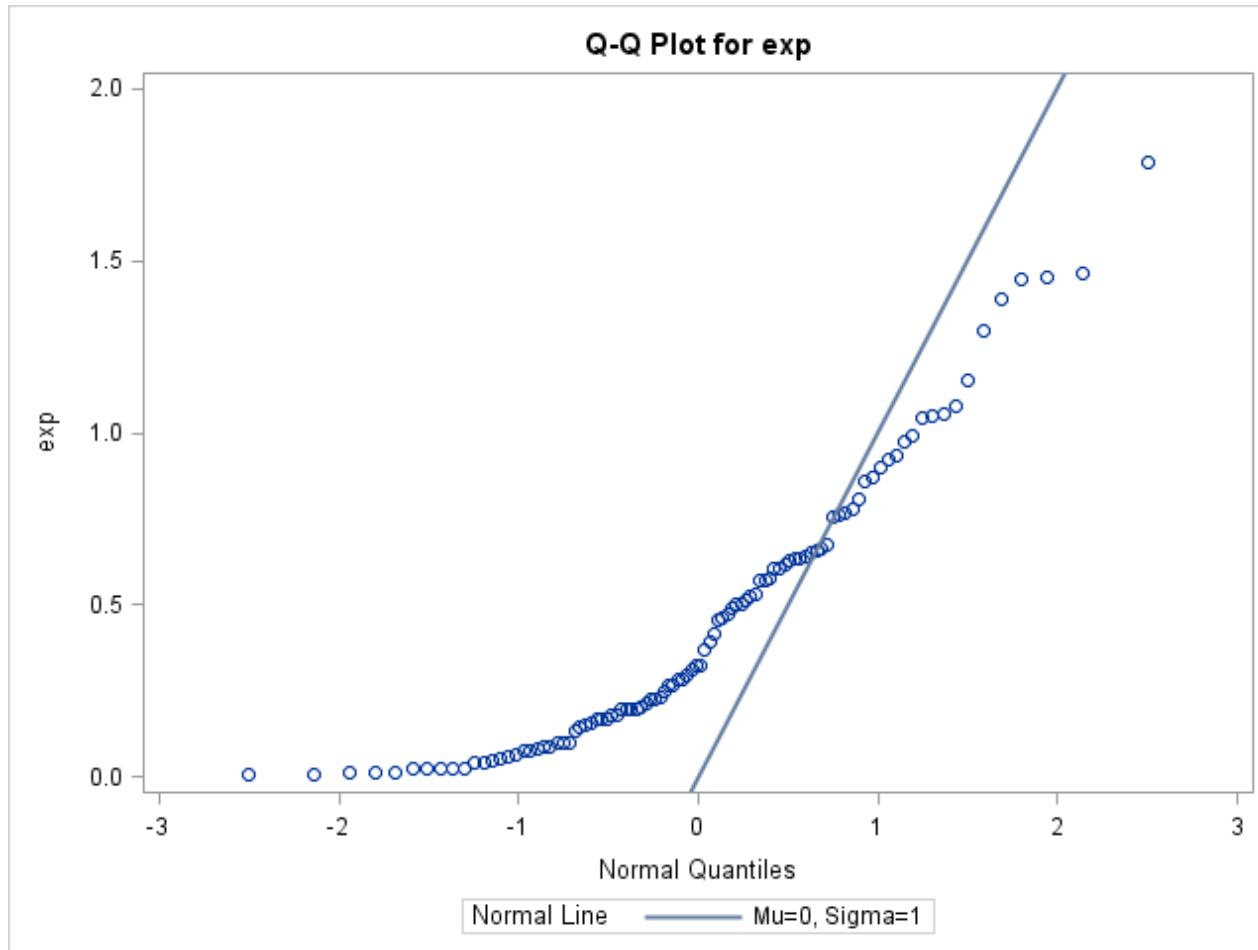
Q-Q Plot for NORMAL1 Assuming a Standard Normal Distribution



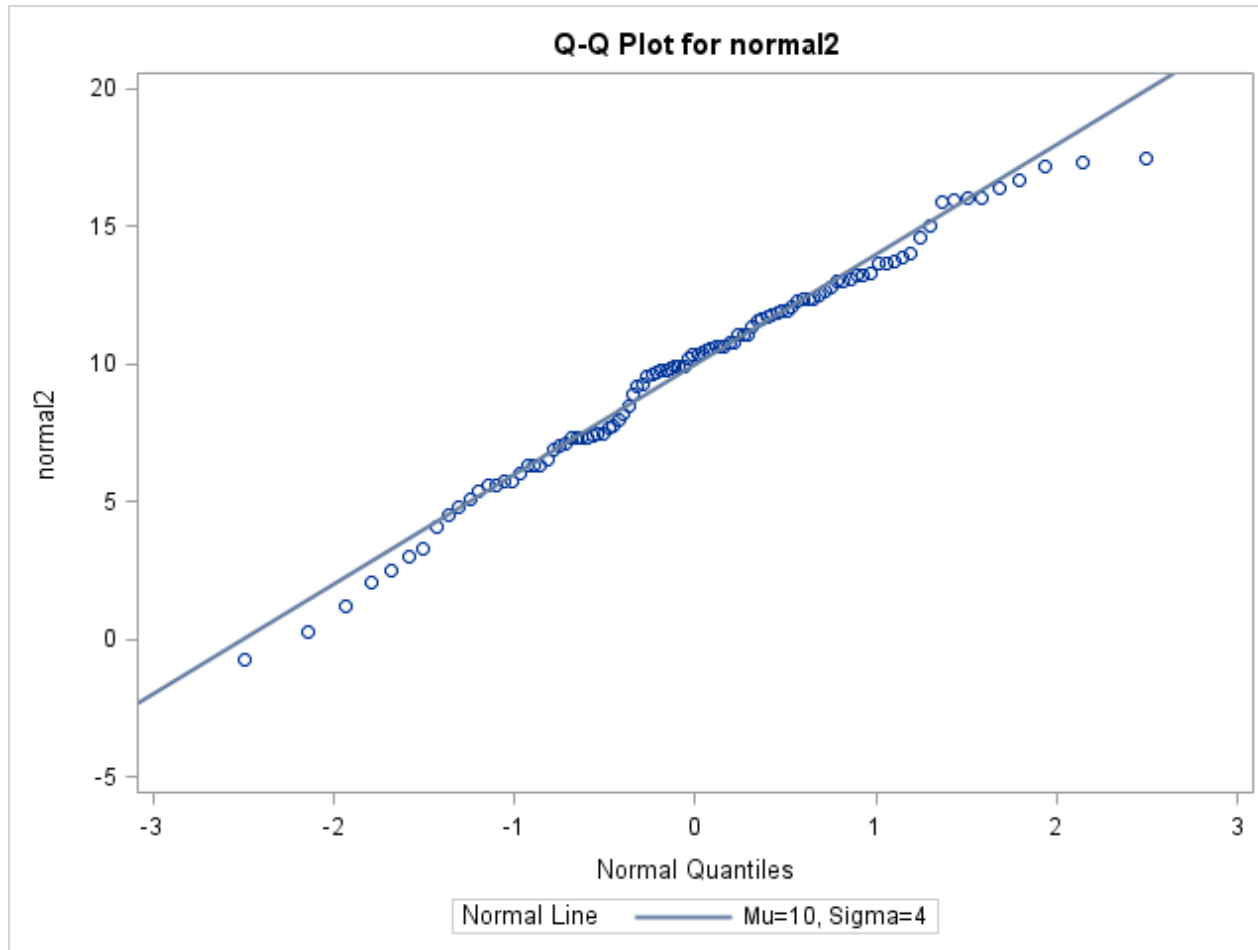
Q-Q Plot for NORMAL2 Assuming a Standard Normal Distribution



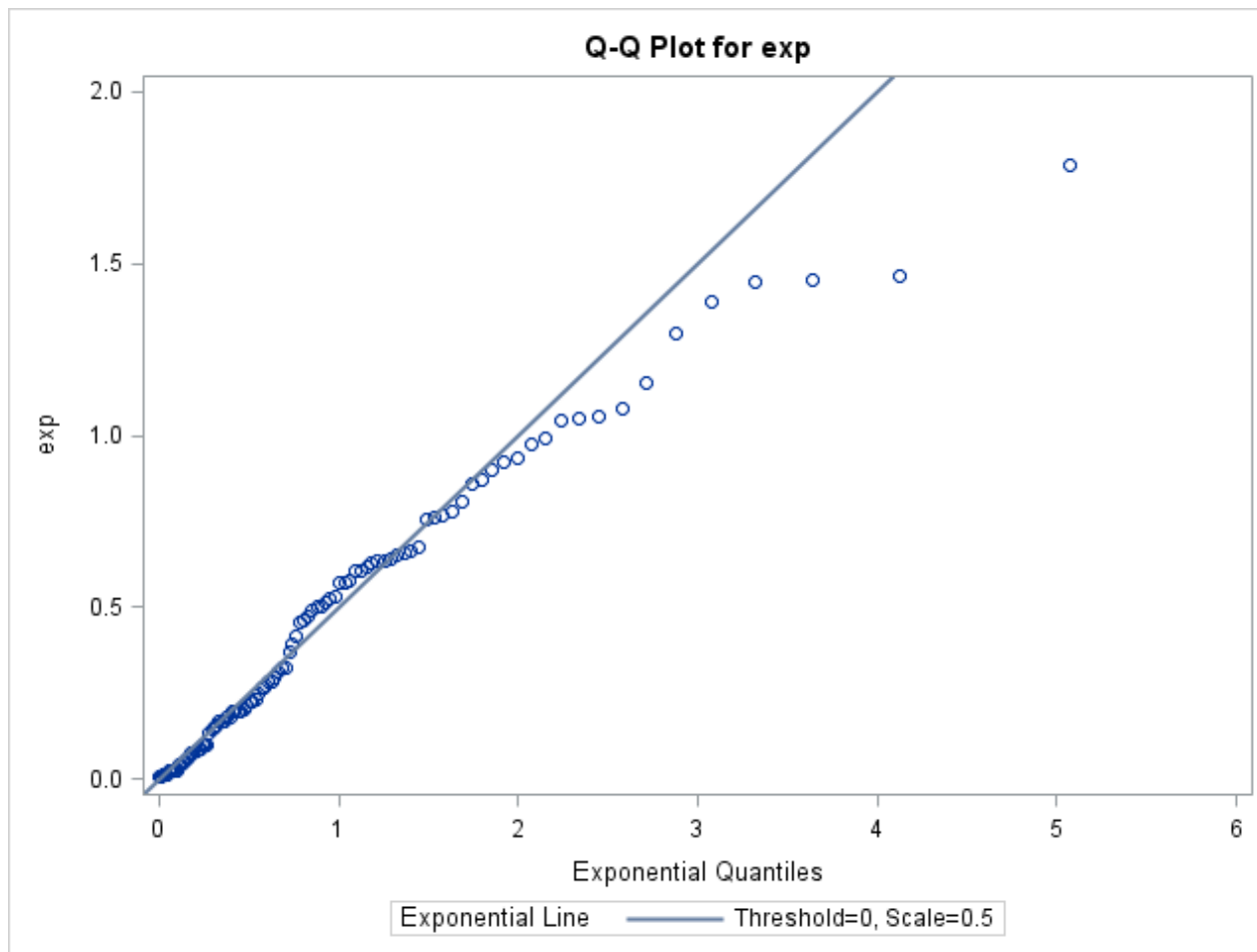
Q-Q Plot for EXP Assuming a Standard Normal Distribution



Q-Q Plot for NORMAL2 Assuming an $N(10,4)$ -Distribution



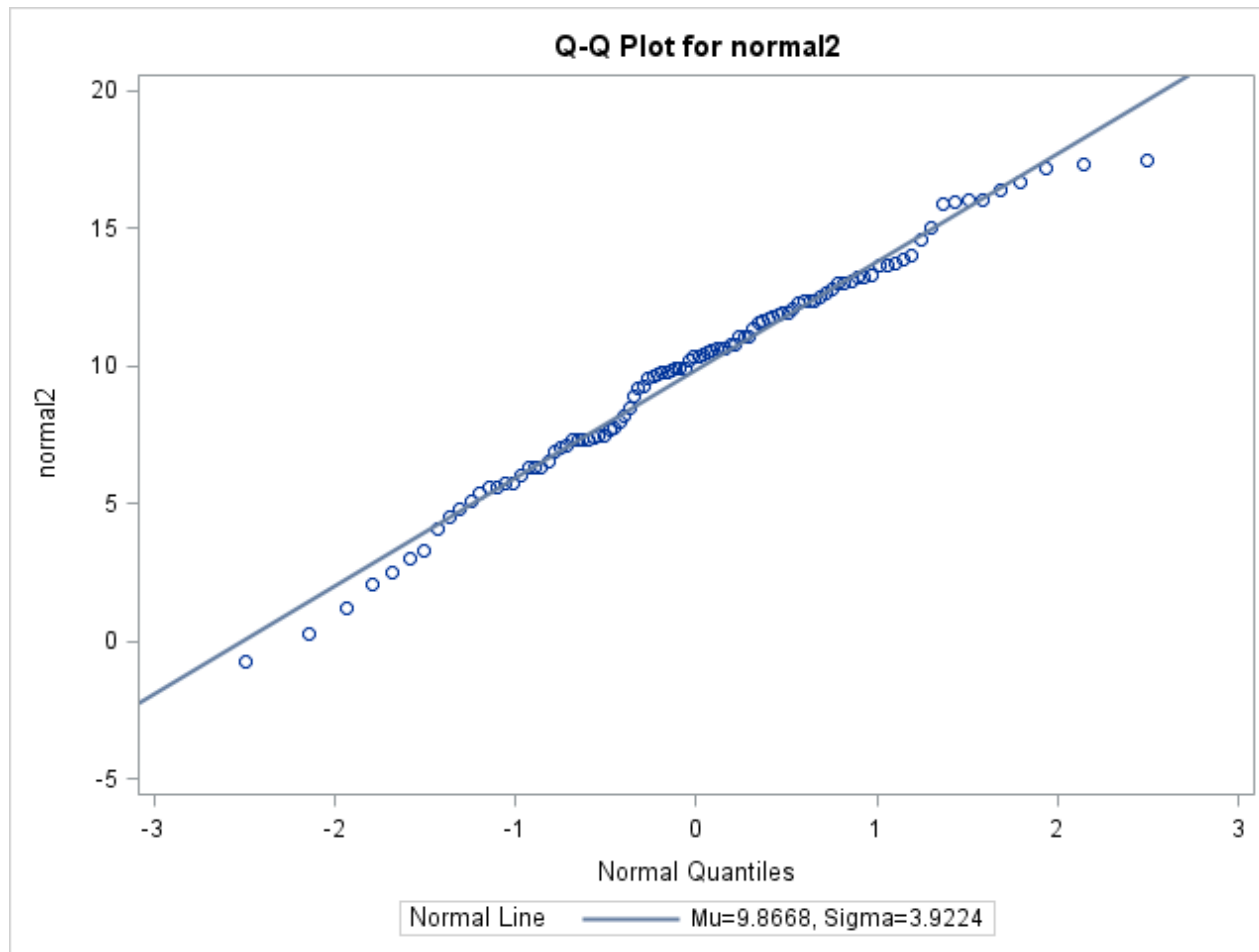
Q-Q Plot for EXP Assuming an Exponential Distribution ($\lambda = 2$)



PROC UNIVARIATE-Call Estimating Parameters

```
proc univariate data=normal noprint;  
    qqplot normal2  
        / normal(mu=est sigma=est) ;  
  
run;
```

Q-Q Plot for NORMAL2 Assuming a Normal Distribution and Estimating Parameters



Introduction

Probability Plots in Theory

Probability Plots in Practice

Q-Q Plot Empirical vs. Empirical

Conclusion

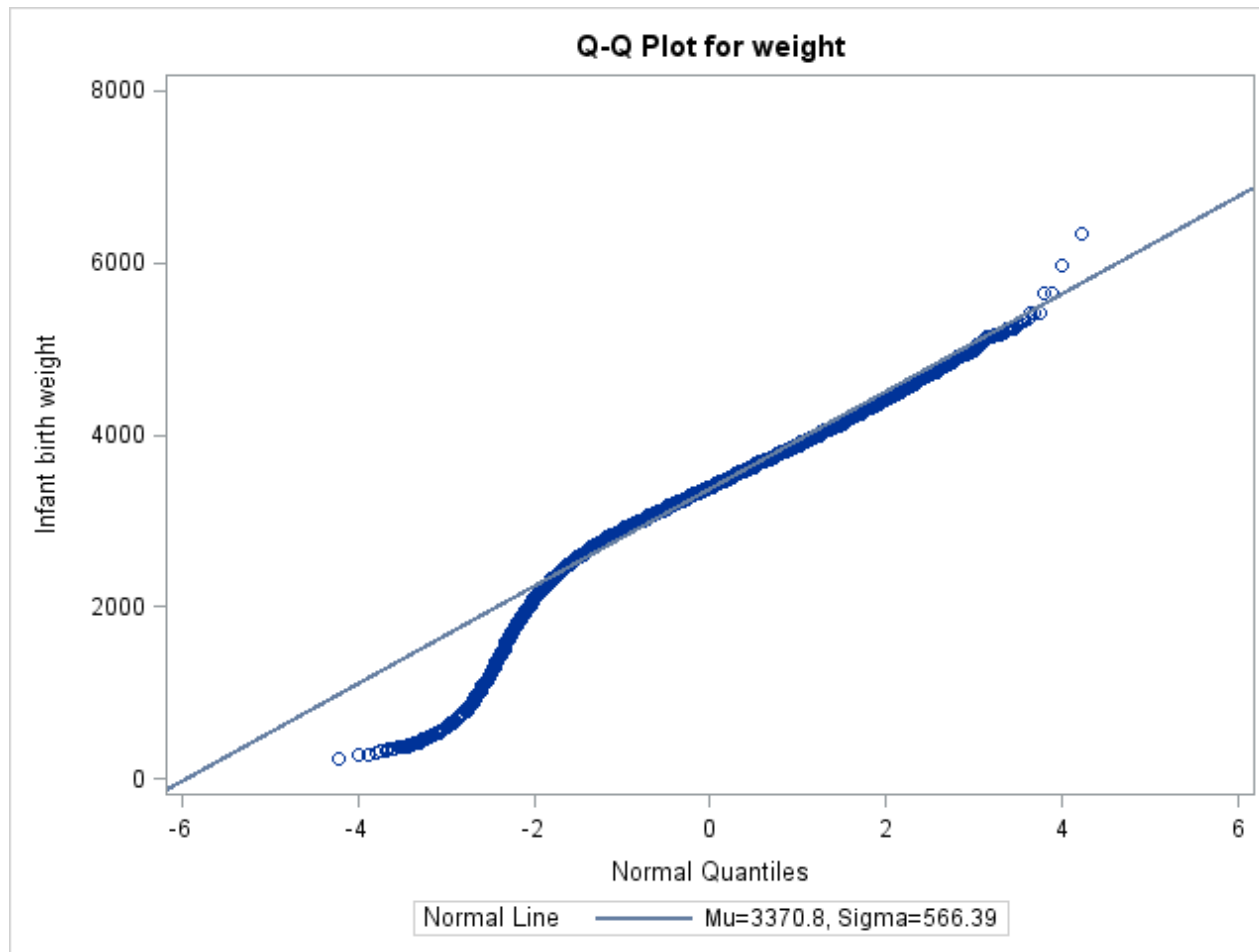
SASHELP.BWEIGHT

- Data of the National Center for Health Statistics
- on the birth weight of 50,000 infants
- in 1997.

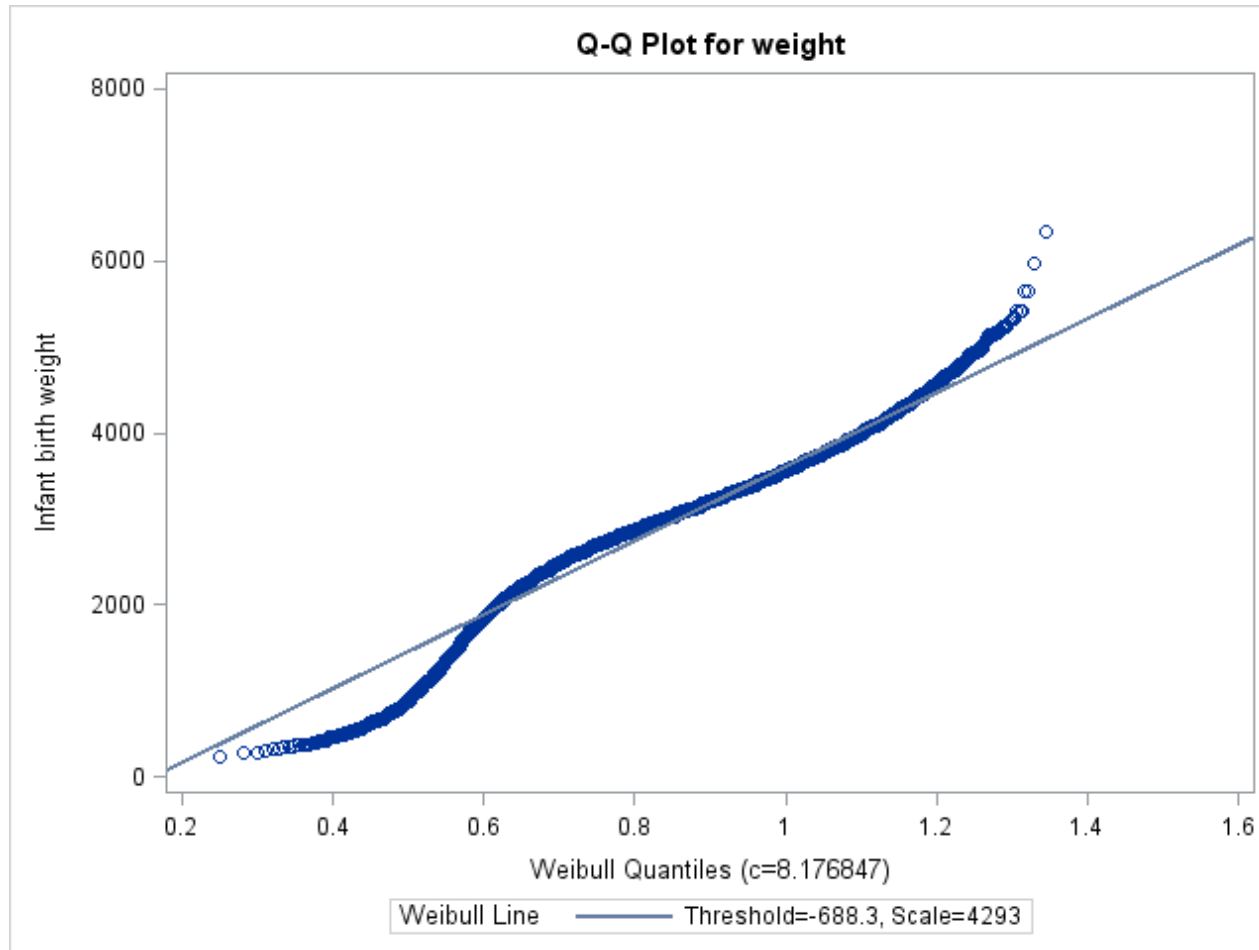
Variables of interest:

- WEIGHT: Infant birth weight
- SMOKE: Mother's smoking status

Q-Q Plot for WEIGHT Assuming a Normal Distribution and Estimating Parameters



Q-Q Plot for WEIGHT Assuming a Weibull Distribution and Estimating Parameters

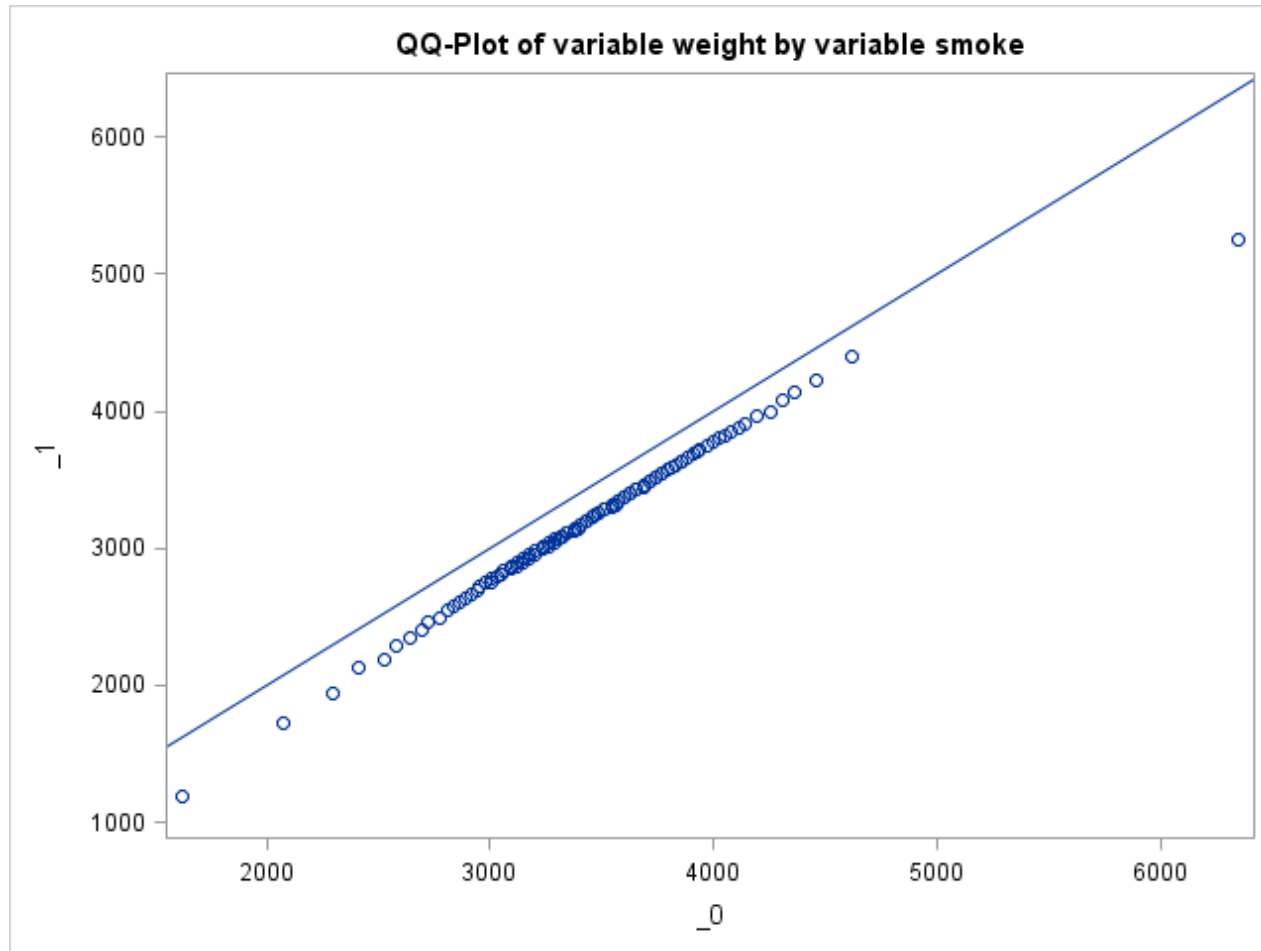


%qq-Macro for Between-Group Comparison

```
%qq(  
    data=sashelp.bweight,  
    var=weight,  
    class=smoke  
);
```

- **data:** input dataset
- **var:** variable to be analyzed
- **class:** binary variable containing group membership

Q-Q Plot for WEIGHT by SMOKE



Introduction

Probability Plots in Theory

Probability Plots in Practice

Q-Q Plot Empirical vs. Empirical

Conclusion

Doing now what patients need next

```
%macro qq(data, var, class);  
    %local ngroups nobs value1 value2 increment;  
  
    proc sort data=&data.;  
        by &class.;  
    proc means data=&data. noprint;  
        var &var.;  
        by &class.;  
        output out=_n(where=(_stat_="N"));  
    run;  
  
    proc sql noprint;  
        select count(*), min(&var.), &class. into :ngroups, :nobs, :value1-:value2  
        from _n;  
    quit;  
  
    %if %sysfunc(anydigit(&value1.)) = 1 %then %let value1 = &_amp;value1.;  
    %if %sysfunc(anydigit(&value2.)) = 1 %then %let value2 = &_amp;value2.;  
  
    %if &ngroups ne 2 %then %do;  
        %put ERROR: Macro qq: &class. does not have two distinct values.;  
        %goto abort_macro;  
    %end;
```

* Choose number of displayed percentiles dependent on number of observations;

```
        %if &nobs. >= 100 %then %let increment = 1;
%else %if &nobs. >= 50 %then %let increment = 2;
%else %if &nobs. >= 25 %then %let increment = 4;
%else %if &nobs. >= 20 %then %let increment = 5;
%else %if &nobs. >= 10 %then %let increment = 10;
%else %if &nobs. >= 5 %then %let increment = 20;
%else %if &nobs. >= 4 %then %let increment = 25;
%else %if &nobs. >= 2 %then %let increment = 50;
%else %if &nobs. >= 1 %then %let increment = 100;
```

* Calculate percentiles;

```
proc univariate data=&data. noprint;
    var &var.;
    class &class.;
    output out=_qql pctlpts=1 to 100 by &increment. pctlpre=p;
    format &class.;
run;
```

```

* Prepare data;

proc transpose data=_qq1 out=_qq2;
    var p;;
    id &class.;
run;

* Plot;

proc sgplot data=_qq2 noautolegend;
    title "QQ-Plot of variable &var. by variable &class.";
    scatter x=&value1. y=&value2.;
    lineparm x=0 y=0 slope=1;
run;

proc datasets library=work;
    delete _n _qq1 _qq2;
run;

%abort_macro:
%mend qq;

```