Empirical software engineering

Lab 3: Design and simulate an experiment
Group 9

Anton Lutteman Daniel Olsson Gerson Silva Filho Johan Mejborn

11 December 2020

OSF experiment registration: https://osf.io/4yxas

Simulate the data

1) Simulate for a the sample size per group that you defined in the power analysis.

a) How does your fitted model looks like?

- **Test Assumption 1 - Homoscedasticity**

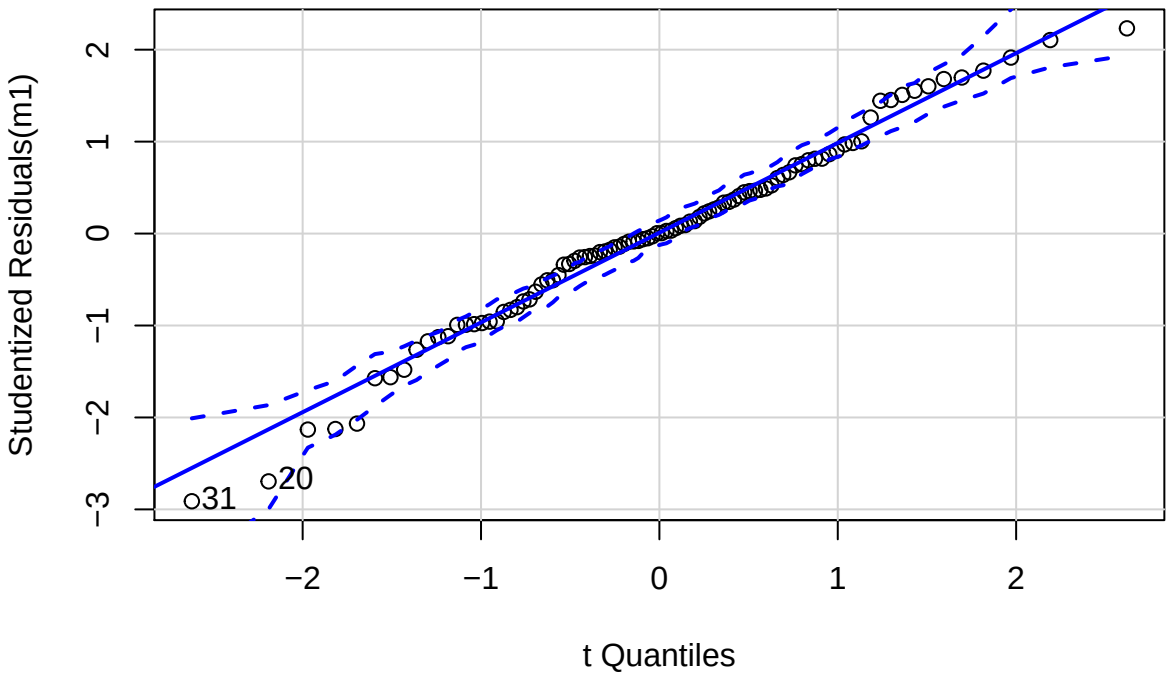
We decided to run a Levene Test to test Homoscedasticity.

```
car::leveneTest(m1)
```

```
## Levene's Test for Homogeneity of Variance (center = median)
##       Df F value Pr(>F)
## group 11  0.7664 0.6721
##       84
```

The result show that we cannot reject homoscedasticity. Then let's also verify the QQ-Plot in order to get another point of view.

```
car::qqPlot(m1)
```




```
## [1] 20 31
```

QQplot shows that we cannot reject that the data is homoscedastic.

- **Test Assumption 2 - Normality**

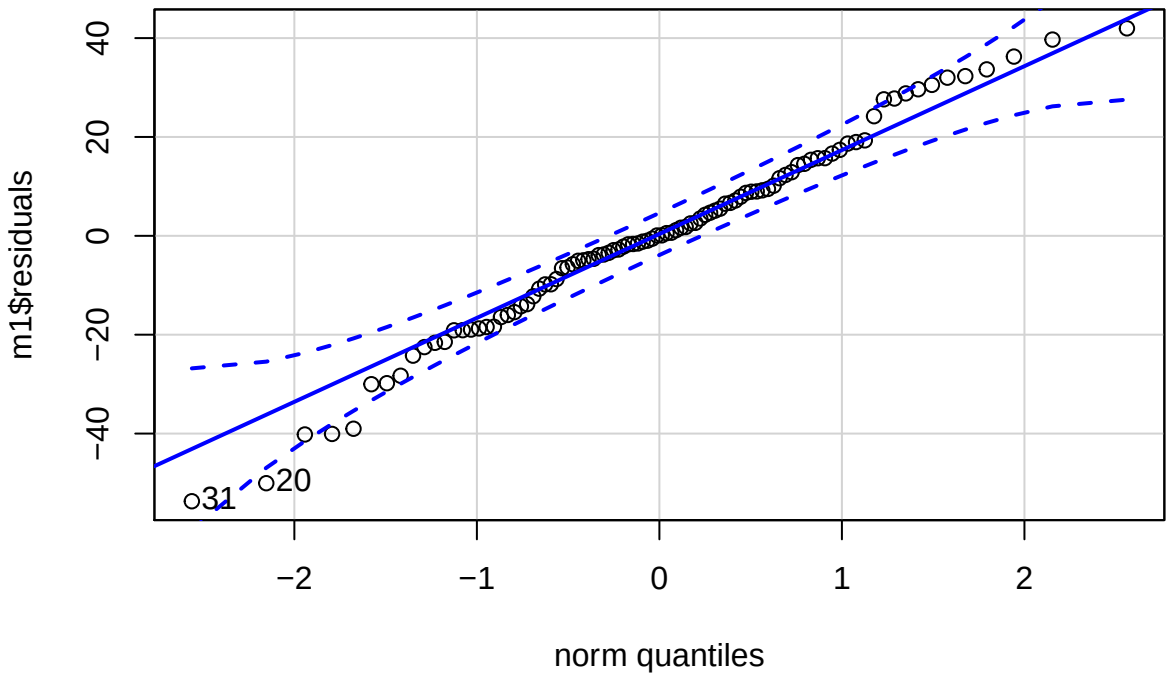
In order to verify normality we run a Shapiro Wilk test.

```
shapiro.test(m1$residuals)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  m1$residuals
## W = 0.98572, p-value = 0.3862
```

The result show that we cannot reject normality. And in order to verify it graphically we used the QQ-plot of the residuals.

```
car::qqPlot(m1$residuals)
```




```
## [1] 31 20
```

The QQplot on the residuals shows normality.

- **Test Assumption 3 - Independence**

Based on how we collected the data, we can assume independence.

- **The analysis**

After all the assumptions have been met, we can run an ANOVA test.

```
car::Anova(m1)
```

```
## Anova Table (Type II tests)
##
## Response: y
##                          Sum Sq Df  F value    Pr(>F)
## Language                 172356  2 203.6018 < 2.2e-16 ***
## IDE                           9  1   0.0208   0.88554
## Experience                42384  1 100.1361 5.630e-16 ***
## Language:IDE              10323  2  12.1942 2.241e-05 ***
## Language:Experience        1497  2   1.7682   0.17694
## IDE:Experience             2375  1   5.6107   0.02015 *
```

```
## Language:IDE:Experience   2274  2   2.6865   0.07397 .
## Residuals                35555 84
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The ANOVA test shows that we can reject that Language, Experience, Language:IDE, Language:Experience and IDE:Experience has the same mean for the levels because of the low p values.

So our conclusions on the experiment hypothesis are:

Hypothesis 1: IDE alone doesn't seem to have an effect on LOC but we can see that it has a significant effect in combination with Language and Experience.

Hypothesis 2: Experience seem to have a significant effect on LOC based on the result.

Hypothesis 3: Language seem to have a significant effect on LOC bases on the result.

Then to get the highest combination for lines of code we can use the Tukey test:

```
TukeyHSD(aov(m1))
```

```
##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
## Fit: aov(formula = m1)
##
## $Language
##                   diff        lwr       upr p adj
## C++-Java      68.57084   56.29898  80.84270     0
## Python-Java  -33.18850  -45.46036 -20.91663     0
## Python-C++  -101.75933 -114.03120 -89.48747     0
##
## $IDE
##                              diff       lwr      upr     p adj
## Visual Studio-Intelij -0.6063779 -8.957631 7.744875 0.8855373
##
## $Experience
##                   diff      lwr      upr p adj
## Senior-Junior 42.02401 33.67276 50.37527     0
##
## $`Language:IDE`
##                                              diff         lwr          upr     p adj
## C++:Intelij-Java:Intelij                    55.48549   34.271060   76.6999153 0.0000000
## Python:Intelij-Java:Intelij                -20.87747  -42.091893    0.3369623 0.0563554
## Java:Visual Studio-Java:Intelij             -1.12259  -22.337018   20.0918375 0.9999874
## C++:Visual Studio-Java:Intelij              80.53360   59.319170  101.7480252 0.0000000
## Python:Visual Studio-Java:Intelij          -46.62212  -67.836546  -25.4076912 0.0000001
## Python:Intelij-C++:Intelij                 -76.36295  -97.577381  -55.1485255 0.0000000
## Java:Visual Studio-C++:Intelij             -56.60808  -77.822505  -35.3936503 0.0000000
## C++:Visual Studio-C++:Intelij               25.04811    3.833682   46.2625374 0.0112133
## Python:Visual Studio-C++:Intelij          -102.10761 -123.322034  -80.8931790 0.0000000
## Java:Visual Studio-Python:Intelij           19.75488   -1.459552   40.9693028 0.0827125
## C++:Visual Studio-Python:Intelij           101.41106   80.196635  122.6254905 0.0000000
## Python:Visual Studio-Python:Intelij        -25.74465  -46.959081   -4.5302259 0.0083412
## C++:Visual Studio-Java:Visual Studio        81.65619   60.441760  102.8706153 0.0000000
## Python:Visual Studio-Java:Visual Studio    -45.49953  -66.713956  -24.2851011 0.0000002
```

```
## Python:Visual Studio-C++:Visual Studio  -127.15572 -148.370144 -105.9412889 0.0000000
##
## $`Language:Experience`
##                                    diff        lwr        upr     p adj
## C++:Junior-Java:Junior         61.841372   40.62694  83.055800 0.0000000
## Python:Junior-Java:Junior     -30.536648  -51.75108  -9.322220 0.0009168
## Java:Senior-Java:Junior        39.305602   18.09117  60.520030 0.0000088
## C++:Senior-Java:Junior        114.605906   93.39148 135.820333 0.0000000
## Python:Senior-Java:Junior       3.465256  -17.74917  24.679683 0.9968543
## Python:Junior-C++:Junior      -92.378020 -113.59245 -71.163592 0.0000000
## Java:Senior-C++:Junior        -22.535770  -43.75020  -1.321342 0.0306839
## C++:Senior-C++:Junior          52.764534   31.55011  73.978961 0.0000000
## Python:Senior-C++:Junior      -58.376116  -79.59054 -37.161689 0.0000000
## Java:Senior-Python:Junior      69.842250   48.62782  91.056677 0.0000000
## C++:Senior-Python:Junior      145.142553  123.92813 166.356981 0.0000000
## Python:Senior-Python:Junior    34.001903   12.78748  55.216331 0.0001582
## C++:Senior-Java:Senior         75.300304   54.08588  96.514731 0.0000000
## Python:Senior-Java:Senior     -35.840346  -57.05477 -14.625919 0.0000595
## Python:Senior-C++:Senior     -111.140650 -132.35508 -89.926222 0.0000000
##
## $`IDE:Experience`
##                                              diff        lwr       upr     p adj
## Visual Studio:Junior-Intelij:Junior       -10.55384 -26.121359  5.013688 0.2915611
## Intelij:Senior-Intelij:Junior              32.07656  16.509031 47.644079 0.0000036
## Visual Studio:Senior-Intelij:Junior        41.41764  25.850111 56.985159 0.0000000
## Intelij:Senior-Visual Studio:Junior        42.63039  27.062867 58.197915 0.0000000
## Visual Studio:Senior-Visual Studio:Junior  51.97147  36.403947 67.538994 0.0000000
## Visual Studio:Senior-Intelij:Senior         9.34108  -6.226444 24.908604 0.3995822
##
## $`Language:IDE:Experience`
##                                                                 diff         lwr         upr     p ad
## C++:Intelij:Junior-Java:Intelij:Junior                     44.8101364   10.226377   79.393895 0.00208
## Python:Intelij:Junior-Java:Intelij:Junior                 -29.9415944  -64.525353    4.642165 0.15620
## Java:Visual Studio:Junior-Java:Intelij:Junior             -21.5112904  -56.095049   13.072469 0.63026
## C++:Visual Studio:Junior-Java:Intelij:Junior               57.3613172   22.777558   91.945076 0.00001
## Python:Visual Studio:Junior-Java:Intelij:Junior           -52.6429914  -87.226750  -18.059232 0.00011
## Java:Intelij:Senior-Java:Intelij:Junior                    18.9169017  -15.666857   53.500661 0.79177
## C++:Intelij:Senior-Java:Intelij:Junior                     85.0777408   50.493982  119.661500 0.00000
## Python:Intelij:Senior-Java:Intelij:Junior                   7.1035655  -27.480194   41.687325 0.99991
## Java:Visual Studio:Senior-Java:Intelij:Junior              38.1830119    3.599253   72.766771 0.01791
## C++:Visual Studio:Senior-Java:Intelij:Junior              122.6227798   88.039021  157.206539 0.00000
## Python:Visual Studio:Senior-Java:Intelij:Junior           -21.6843445  -56.268104   12.899415 0.61861
## Python:Intelij:Junior-C++:Intelij:Junior                  -74.7517308 -109.335490  -40.167972 0.00000
## Java:Visual Studio:Junior-C++:Intelij:Junior              -66.3214268 -100.905186  -31.737668 0.00000
## C++:Visual Studio:Junior-C++:Intelij:Junior                12.5511807  -22.032578   47.134940 0.98585
## Python:Visual Studio:Junior-C++:Intelij:Junior            -97.4531278 -132.036887  -62.869369 0.00000
## Java:Intelij:Senior-C++:Intelij:Junior                    -25.8932348  -60.476994    8.690524 0.34417
## C++:Intelij:Senior-C++:Intelij:Junior                      40.2676044    5.683845   74.851363 0.00941
## Python:Intelij:Senior-C++:Intelij:Junior                  -37.7065709  -72.290330   -3.122812 0.02065
## Java:Visual Studio:Senior-C++:Intelij:Junior               -6.6271245  -41.210884   27.956635 0.99995
## C++:Visual Studio:Senior-C++:Intelij:Junior                77.8126434   43.228884  112.396402 0.00000
## Python:Visual Studio:Senior-C++:Intelij:Junior            -66.4944809 -101.078240  -31.910722 0.00000
## Java:Visual Studio:Junior-Python:Intelij:Junior             8.4303040  -26.153455   43.014063 0.99956
## C++:Visual Studio:Junior-Python:Intelij:Junior            87.3029116   52.719153  121.886671 0.00000
```

```
## Python:Visual Studio:Junior-Python:Intelij:Junior        -22.7013970  -57.285156   11.882362 0.549539
## Java:Intelij:Senior-Python:Intelij:Junior                 48.8584961   14.274737   83.442255 0.000490
## C++:Intelij:Senior-Python:Intelij:Junior                 115.0193352   80.435576  149.603094 0.000000
## Python:Intelij:Senior-Python:Intelij:Junior               37.0451599    2.461401   71.628919 0.025085
## Java:Visual Studio:Senior-Python:Intelij:Junior           68.1246063   33.540847  102.708365 0.000000
## C++:Visual Studio:Senior-Python:Intelij:Junior           152.5643742  117.980615  187.148133 0.000000
## Python:Visual Studio:Senior-Python:Intelij:Junior          8.2572499  -26.326509   42.841009 0.999644
## C++:Visual Studio:Junior-Java:Visual Studio:Junior        78.8726076   44.288849  113.456367 0.000000
## Python:Visual Studio:Junior-Java:Visual Studio:Junior    -31.1317010  -65.715460    3.452058 0.119238
## Java:Intelij:Senior-Java:Visual Studio:Junior             40.4281921    5.844433   75.011951 0.008951
## C++:Intelij:Senior-Java:Visual Studio:Junior             106.5890312   72.005272  141.172790 0.000000
## Python:Intelij:Senior-Java:Visual Studio:Junior           28.6148559   -5.968903   63.198615 0.206995
## Java:Visual Studio:Senior-Java:Visual Studio:Junior       59.6943023   25.110543   94.278061 0.000007
## C++:Visual Studio:Senior-Java:Visual Studio:Junior       144.1340702  109.550311  178.717829 0.000000
## Python:Visual Studio:Senior-Java:Visual Studio:Junior     -0.1730541  -34.756813   34.410705 1.000000
## Python:Visual Studio:Junior-C++:Visual Studio:Junior    -110.0043085 -144.588068  -75.420549 0.000000
## Java:Intelij:Senior-C++:Visual Studio:Junior            -38.4444155  -73.028175   -3.860656 0.016559
## C++:Intelij:Senior-C++:Visual Studio:Junior              27.7164236   -6.867335   62.300183 0.247425
## Python:Intelij:Senior-C++:Visual Studio:Junior          -50.2577517  -84.841511  -15.673993 0.000291
## Java:Visual Studio:Senior-C++:Visual Studio:Junior      -19.1783053  -53.762064   15.405454 0.777105
## C++:Visual Studio:Senior-C++:Visual Studio:Junior        65.2614626   30.677704   99.845222 0.000000
## Python:Visual Studio:Senior-C++:Visual Studio:Junior     -79.0456617 -113.629421  -44.461903 0.000000
## Java:Intelij:Senior-Python:Visual Studio:Junior          71.5598930   36.976134  106.143652 0.000000
## C++:Intelij:Senior-Python:Visual Studio:Junior          137.7207322  103.136973  172.304491 0.000000
## Python:Intelij:Senior-Python:Visual Studio:Junior        59.7465569   25.162798   94.330316 0.000007
## Java:Visual Studio:Senior-Python:Visual Studio:Junior    90.8260033   56.242244  125.409762 0.000000
## C++:Visual Studio:Senior-Python:Visual Studio:Junior    175.2657712  140.682012  209.849530 0.000000
## Python:Visual Studio:Senior-Python:Visual Studio:Junior  30.9586469   -3.625112   65.542406 0.124137
## C++:Intelij:Senior-Java:Intelij:Senior                   66.1608391   31.577080  100.744598 0.000000
## Python:Intelij:Senior-Java:Intelij:Senior               -11.8133362  -46.397095   22.770423 0.991305
## Java:Visual Studio:Senior-Java:Intelij:Senior            19.2661102  -15.317649   53.849869 0.772070
## C++:Visual Studio:Senior-Java:Intelij:Senior            103.7058781   69.122119  138.289637 0.000000
## Python:Visual Studio:Senior-Java:Intelij:Senior         -40.6012461  -75.185005   -6.017487 0.008471
## Python:Intelij:Senior-C++:Intelij:Senior                -77.9741753 -112.557934  -43.390416 0.000000
## Java:Visual Studio:Senior-C++:Intelij:Senior            -46.8947289  -81.478488  -12.310970 0.001001
## C++:Visual Studio:Senior-C++:Intelij:Senior              37.5450390    2.961280   72.128798 0.021660
## Python:Visual Studio:Senior-C++:Intelij:Senior         -106.7620853 -141.345844  -72.178326 0.000000
## Java:Visual Studio:Senior-Python:Intelij:Senior          31.0794464   -3.504313   65.663205 0.120701
## C++:Visual Studio:Senior-Python:Intelij:Senior          115.5192143   80.935455  150.102973 0.000000
## Python:Visual Studio:Senior-Python:Intelij:Senior       -28.7879100  -63.371669    5.795849 0.199771
## C++:Visual Studio:Senior-Java:Visual Studio:Senior       84.4397679   49.856009  119.023527 0.000000
## Python:Visual Studio:Senior-Java:Visual Studio:Senior   -59.8673564  -94.451115  -25.283597 0.000000
## Python:Visual Studio:Senior-C++:Visual Studio:Senior    -144.3071243 -178.890883 -109.723365 0.000000
```

Without drawing conclusions for all combinations presented by the Tukey test we can say that the highest combination for lines of code is **Language = C++, IDE = Visual Studio and Experience = Senior**.

b) How do it compare with your 'true' model you defined in the simulation.

When we compare the created model defined in our main function with the defined coefficients, with the data generated by the model, we can see that they are similar. Also we can notice some variations that can be explained either by the defined standard deviation of the model, or that the model interpreted the data in another way than what we set up. The difference in the first order is very small compared to our true model, however the third order interactions are deviating quite a bit compared to our true model.

We can observe the differences looking in the following table and the defined factors:

	Defined Coeficients	Model Coeficients
(Intercept)	100	100.703339
LanguageC++	50	44.810136
LanguagePython	-30	-29.941594
IDEVisual Studio	-10	-21.511290
ExperienceSenior	30	18.916902
LanguageC++:IDEVisual Studio	20	34.062471
LanguagePython:IDEVisual Studio	-20	-1.190107
LanguageC++:ExperienceSenior	10	21.350703
LanguagePython:ExperienceSenior	10	18.128258
IDEVisual Studio:ExperienceSenior	20	40.777401
LanguageC++:IDEVisual Studio:ExperienceSenior	10	-15.783542
LanguagePython:IDEVisual Studio:ExperienceSenior	-15	-46.863914

2) Simulate for an underpowered study with half the sample size you calculated in the power analysis

a) How does your fitted model looks like?

- **Test Assumption 1 - Homoscedasticity**

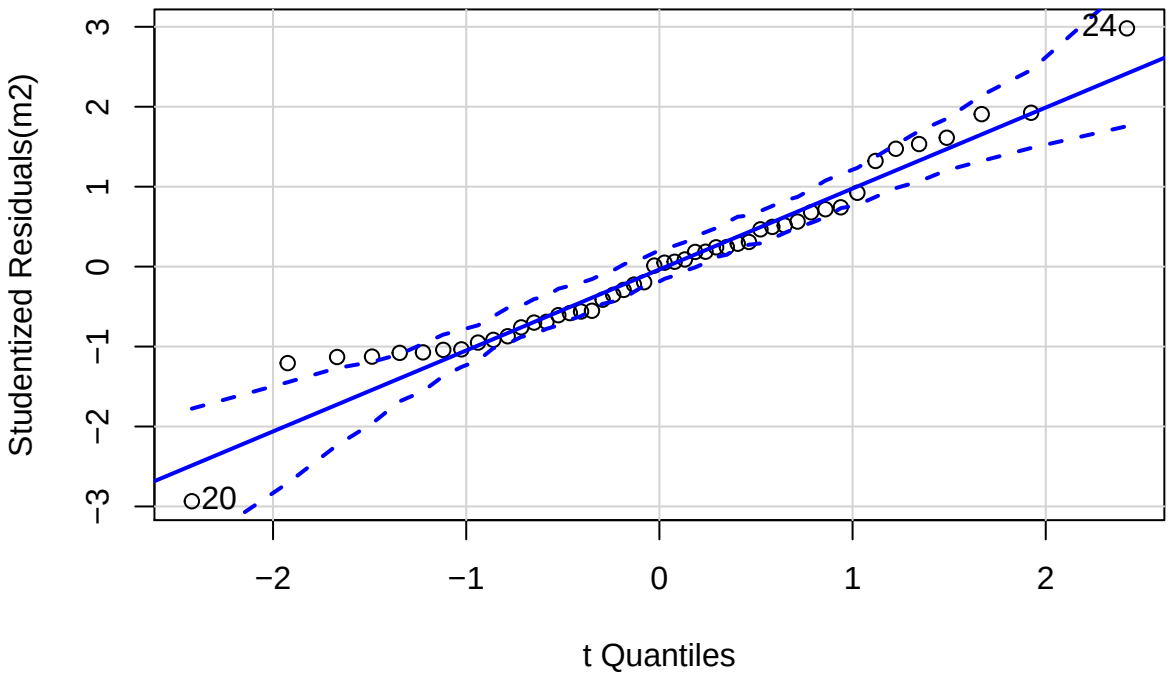
We decided to run a Levene Test to test Homoscedasticity.

```
car::leveneTest(m2)
```

```
## Levene's Test for Homogeneity of Variance (center = median)
##       Df F value Pr(>F)
## group 11     0.4 0.9469
##       36
```

The result show that we cannot reject homoscedasticity. Then let's also verify the QQ-Plot in order to get another point of view.

```
car::qqPlot(m2)
```




```
## [1] 20 24
```

QQplot shows that we cannot reject that the data is homoscedastic.

- **Test Assumption 2 - Normality**

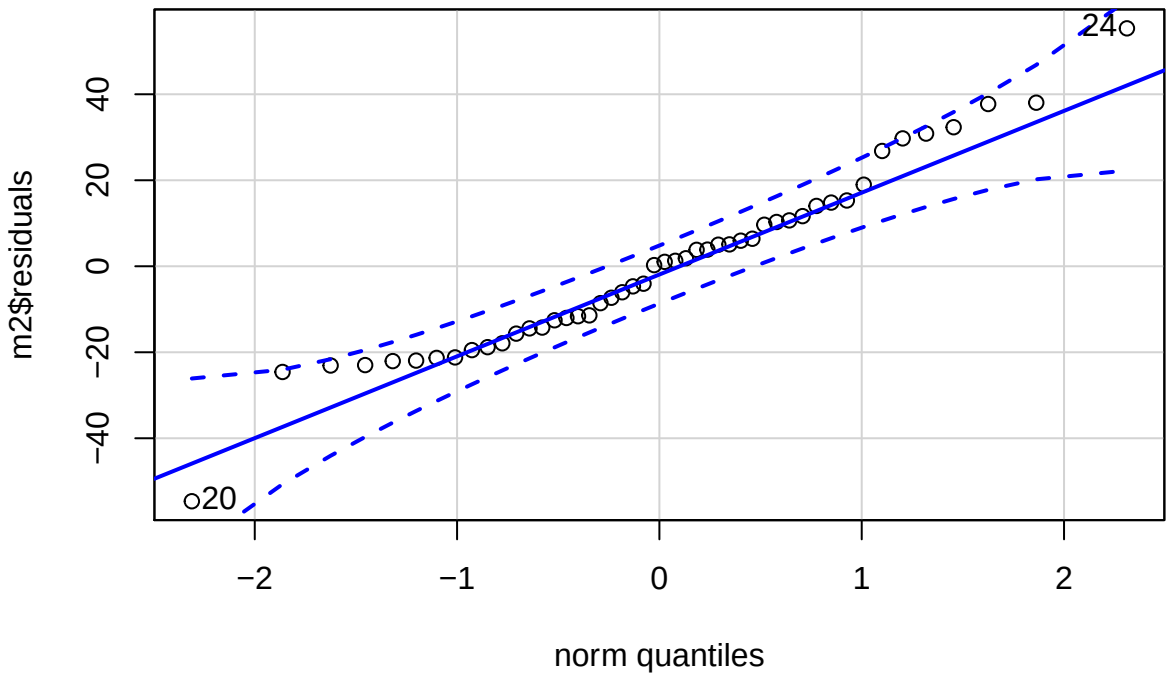
In order to verify normality we run a Shapiro Wilk test.

```
shapiro.test(m2$residuals)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  m2$residuals
## W = 0.96842, p-value = 0.2198
```

The result show that we cannot reject normality. And in order to verify it graphically we used the QQ-plot of the residuals.

```
car::qqPlot(m2$residuals)
```




```
## [1] 24 20
```

The QQplot on the residuals shows normality.

- **Test Assumption 3 - Independence**

Based on how we collected the data, we can assume independence.

- **The analysis**

After all the assumptions have been met, we can run an ANOVA test.

```
car::Anova(m2)
```

```
## Anova Table (Type II tests)
##
## Response: y
##                          Sum Sq Df F value    Pr(>F)
## Language                  73167  2 65.3258 1.049e-12 ***
## IDE                         208  1  0.3718   0.54588
## Experience                25459  1 45.4617 7.163e-08 ***
## Language:IDE               2203  2  1.9672   0.15460
## Language:Experience         536  2  0.4782   0.62377
## IDE:Experience               53  1  0.0948   0.75999
```

```
## Language:IDE:Experience   2794  2  2.4948   0.09667 .
## Residuals                20161 36
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The ANOVA test shows that we can reject that Language, Experience, Language:IDE, Language:Experience and IDE:Experience has the same mean for the levels because of the low p values.

So our conclusions on the experiment hypothesis are:

Hypothesis 1: IDE doesn't seem to have an effect on LOC neither alone nor as an interaction.

Hypothesis 2: Experience seem to have a significant effect on LOC based on the result.

Hypothesis 3: Language seem to have a significant effect on LOC bases on the result.

Then to get the highest combination for lines of code we can use the Tukey test:

```
TukeyHSD(aov(m2))
```

```
##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
## Fit: aov(formula = m2)
##
## $Language
##                   diff        lwr       upr     p adj
## C++-Java     60.22071   39.76993  80.67149 0.0000001
## Python-Java -34.22880  -54.67958 -13.77803 0.0006615
## Python-C++  -94.44951 -114.90029 -73.99874 0.0000000
##
## $IDE
##                             diff       lwr      upr     p adj
## Visual Studio-Intelij -4.165246 -18.01998 9.689492 0.5458768
##
## $Experience
##                    diff      lwr     upr p adj
## Senior-Junior 46.06096 32.20623 59.9157 1e-07
##
## $`Language:IDE`
##                                                diff        lwr        upr     p adj
## C++:Intelij-Java:Intelij                   56.196651   20.59815  91.795148 0.0004331
## Python:Intelij-Java:Intelij               -22.297516  -57.89601  13.300980 0.4277062
## Java:Visual Studio-Java:Intelij             1.106240  -34.49226  36.704737 0.9999989
## C++:Visual Studio-Java:Intelij             65.351009   29.75251 100.949506 0.0000421
## Python:Visual Studio-Java:Intelij         -45.053853  -80.65235  -9.455356 0.0064587
## Python:Intelij-C++:Intelij                -78.494167 -114.09266 -42.895670 0.0000014
## Java:Visual Studio-C++:Intelij            -55.090410  -90.68891 -19.491914 0.0005713
## C++:Visual Studio-C++:Intelij               9.154359  -26.44414  44.752856 0.9702429
## Python:Visual Studio-C++:Intelij         -101.250503 -136.84900 -65.652007 0.0000000
## Java:Visual Studio-Python:Intelij          23.403757  -12.19474  59.002254 0.3742992
## C++:Visual Studio-Python:Intelij           87.648526   52.05003 123.247023 0.0000001
## Python:Visual Studio-Python:Intelij       -22.756336  -58.35483  12.842161 0.4051585
## C++:Visual Studio-Java:Visual Studio       64.244769   28.64627  99.843266 0.0000560
## Python:Visual Studio-Java:Visual Studio   -46.160093  -81.75859 -10.561596 0.0049930
## Python:Visual Studio-C++:Visual Studio   -110.404862 -146.00336 -74.806365 0.0000000
```

```
##
## $`Language:Experience`
##                                     diff         lwr        upr     p adj
## C++:Junior-Java:Junior         55.755778   20.157281  91.354275 0.0004837
## Python:Junior-Java:Junior     -30.523109  -66.121606   5.075388 0.1286737
## Java:Senior-Java:Junior        45.554807    9.956310  81.153304 0.0057503
## C++:Senior-Java:Junior        110.240448   74.641951 145.838945 0.0000000
## Python:Senior-Java:Junior       7.620307  -27.978190  43.218804 0.9867255
## Python:Junior-C++:Junior      -86.278888 -121.877385 -50.680391 0.0000002
## Java:Senior-C++:Junior        -10.200971  -45.799468  25.397526 0.9530337
## C++:Senior-C++:Junior          54.484670   18.886173  90.083167 0.0006645
## Python:Senior-C++:Junior      -48.135472  -83.733968 -12.536975 0.0031308
## Java:Senior-Python:Junior      76.077916   40.479419 111.676413 0.0000027
## C++:Senior-Python:Junior      140.763558  105.165061 176.362055 0.0000000
## Python:Senior-Python:Junior    38.143416    2.544919  73.741913 0.0297479
## C++:Senior-Java:Senior         64.685641   29.087144 100.284138 0.0000500
## Python:Senior-Java:Senior     -37.934500  -73.532997  -2.336003 0.0310743
## Python:Senior-C++:Senior     -102.620142 -138.218639 -67.021645 0.0000000
##
## $`IDE:Experience`
##                                                diff       lwr      upr     p adj
## Visual Studio:Junior-Intelij:Junior       -6.268129 -32.28759 19.75133 0.9152787
## Intelij:Senior-Intelij:Junior             43.958081  17.93862 69.97754 0.0003308
## Visual Studio:Senior-Intelij:Junior       41.895719  15.87626 67.91517 0.0006212
## Intelij:Senior-Visual Studio:Junior       50.226210  24.20675 76.24567 0.0000469
## Visual Studio:Senior-Visual Studio:Junior 48.163847  22.14439 74.18330 0.0000896
## Visual Studio:Senior-Intelij:Senior       -2.062363 -28.08182 23.95709 0.9964990
##
## $`Language:IDE:Experience`
##                                                                diff          lwr        upr     p adj
## C++:Intelij:Junior-Java:Intelij:Junior                     58.766665    0.3614939 117.171835 0.0474533
## Python:Intelij:Junior-Java:Intelij:Junior                 -30.069280  -88.4744505  28.335891 0.8090571
## Java:Visual Studio:Junior-Java:Intelij:Junior              -3.958318  -62.3634890  54.446852 1.0000000
## C++:Visual Studio:Junior-Java:Intelij:Junior               48.786574   -9.6185971 107.191744 0.1780983
## Python:Visual Studio:Junior-Java:Intelij:Junior           -34.935257  -93.3404279  23.469913 0.6339208
## Java:Intelij:Senior-Java:Intelij:Junior                    40.490248  -17.9149223  98.895419 0.4193098
## C++:Intelij:Senior-Java:Intelij:Junior                     94.116885   35.7117145 152.522056 0.0001253
## Python:Intelij:Senior-Java:Intelij:Junior                  25.964495  -32.4406755  84.369666 0.9149537
## Java:Visual Studio:Senior-Java:Intelij:Junior              46.661047  -11.7441236 105.066218 0.2272353
## C++:Visual Studio:Senior-Java:Intelij:Junior              122.405693   64.0005225 180.810864 0.0000008
## Python:Visual Studio:Senior-Java:Intelij:Junior           -14.682200  -73.0873707  43.722971 0.9989709
## Python:Intelij:Junior-C++:Intelij:Junior                  -88.835944 -147.2411151 -30.430774 0.0003225
## Java:Visual Studio:Junior-C++:Intelij:Junior              -62.724983 -121.1301536  -4.319812 0.0263075
## C++:Visual Studio:Junior-C++:Intelij:Junior               -9.980091  -68.3852617  48.425080 0.9999749
## Python:Visual Studio:Junior-C++:Intelij:Junior            -93.701922 -152.1070925 -35.296751 0.0001350
## Java:Intelij:Senior-C++:Intelij:Junior                    -18.276416  -76.6815869  40.128754 0.9931709
## C++:Intelij:Senior-C++:Intelij:Junior                      35.350221  -23.0549501  93.755391 0.6176714
## Python:Intelij:Senior-C++:Intelij:Junior                  -32.802169  -91.2073401  25.603001 0.7152037
## Java:Visual Studio:Senior-C++:Intelij:Junior              -12.105617  -70.5107881  46.299553 0.9998313
## C++:Visual Studio:Senior-C++:Intelij:Junior                63.639029    5.2338579 122.044199 0.0228611
## Python:Visual Studio:Senior-C++:Intelij:Junior            -73.448865 -131.8540353 -15.043694 0.0046809
## Java:Visual Studio:Junior-Python:Intelij:Junior            26.110961  -32.2942092  84.516132 0.9120159
## C++:Visual Studio:Junior-Python:Intelij:Junior             78.855853   20.4506827 137.261024 0.0018633
## Python:Visual Studio:Junior-Python:Intelij:Junior          -4.865977  -63.2711481  53.539193 1.0000000
```

```
## Java:Intelij:Senior-Python:Intelij:Junior                70.559528   12.1543575 128.964699 0.007566
## C++:Intelij:Senior-Python:Intelij:Junior                124.186165   65.7809943 182.591336 0.000000
## Python:Intelij:Senior-Python:Intelij:Junior              56.033775   -2.3713957 114.438946 0.069940
## Java:Visual Studio:Senior-Python:Intelij:Junior          76.730327   18.3251562 135.135498 0.002684
## C++:Visual Studio:Senior-Python:Intelij:Junior          152.474973   94.0698023 210.880144 0.000000
## Python:Visual Studio:Senior-Python:Intelij:Junior        15.387080  -43.0180909  73.792250 0.998431
## C++:Visual Studio:Junior-Java:Visual Studio:Junior       52.744892   -5.6602787 111.150063 0.108871
## Python:Visual Studio:Junior-Java:Visual Studio:Junior   -30.976939  -89.3821095  27.428232 0.779585
## Java:Intelij:Senior-Java:Visual Studio:Junior            44.448567  -13.9566040 102.853737 0.287907
## C++:Intelij:Senior-Java:Visual Studio:Junior             98.075204   39.6700328 156.480374 0.000061
## Python:Intelij:Senior-Java:Visual Studio:Junior          29.922814  -28.4823571  88.327984 0.813629
## Java:Visual Studio:Senior-Java:Visual Studio:Junior      50.619365   -7.7858052 109.024536 0.142644
## C++:Visual Studio:Senior-Java:Visual Studio:Junior      126.364012   67.9588409 184.769182 0.000000
## Python:Visual Studio:Senior-Java:Visual Studio:Junior   -10.723882  -69.1290524  47.681289 0.999948
## Python:Visual Studio:Junior-C++:Visual Studio:Junior    -83.721831 -142.1270015 -25.316660 0.000797
## Java:Intelij:Senior-C++:Visual Studio:Junior            -8.296325  -66.7014959  50.108845 0.999996
## C++:Intelij:Senior-C++:Visual Studio:Junior              45.330312  -13.0748591 103.735482 0.262561
## Python:Intelij:Senior-C++:Visual Studio:Junior          -22.822078  -81.2272491  35.583092 0.963244
## Java:Visual Studio:Senior-C++:Visual Studio:Junior       -2.125526  -60.5306972  56.279644 1.000000
## C++:Visual Studio:Senior-C++:Visual Studio:Junior        73.619120   15.2139489 132.024290 0.004549
## Python:Visual Studio:Senior-C++:Visual Studio:Junior    -63.468774 -121.8739443  -5.063603 0.023469
## Java:Intelij:Senior-Python:Visual Studio:Junior          75.425506   17.0203349 133.830676 0.003352
## C++:Intelij:Senior-Python:Visual Studio:Junior          129.052142   70.6469717 187.457313 0.000000
## Python:Intelij:Senior-Python:Visual Studio:Junior        60.899752    2.4945817 119.304923 0.034663
## Java:Visual Studio:Senior-Python:Visual Studio:Junior    81.596304   23.1911337 140.001475 0.001157
## C++:Visual Studio:Senior-Python:Visual Studio:Junior    157.340950   98.9357797 215.746121 0.000000
## Python:Visual Studio:Senior-Python:Visual Studio:Junior  20.253057  -38.1521135  78.658228 0.984615
## C++:Intelij:Senior-Java:Intelij:Senior                   53.626637   -4.7785339 112.031807 0.096962
## Python:Intelij:Senior-Java:Intelij:Senior               -14.525753  -72.9309238  43.879418 0.999066
## Java:Visual Studio:Senior-Java:Intelij:Senior             6.170799  -52.2343719  64.575969 0.999998
## C++:Visual Studio:Senior-Java:Intelij:Senior             81.915445   23.5102742 140.320616 0.001094
## Python:Visual Studio:Senior-Java:Intelij:Senior         -55.172448 -113.5776191   3.232722 0.078746
## Python:Intelij:Senior-C++:Intelij:Senior                -68.152390 -126.5575606  -9.747219 0.011202
## Java:Visual Studio:Senior-C++:Intelij:Senior            -47.455838 -105.8610087  10.949333 0.207821
## C++:Visual Studio:Senior-C++:Intelij:Senior              28.288808  -30.1163626  86.693979 0.860851
## Python:Visual Studio:Senior-C++:Intelij:Senior         -108.799085 -167.2042559 -50.393914 0.000008
## Java:Visual Studio:Senior-Python:Intelij:Senior          20.696552  -37.7086187  79.101723 0.981870
## C++:Visual Studio:Senior-Python:Intelij:Senior           96.441198   38.0360273 154.846369 0.000082
## Python:Visual Studio:Senior-Python:Intelij:Senior       -40.646695  -99.0518659  17.758475 0.413632
## C++:Visual Studio:Senior-Java:Visual Studio:Senior       75.744646   17.3394754 134.149817 0.003175
## Python:Visual Studio:Senior-Java:Visual Studio:Senior   -61.343247 -119.7484178  -2.938076 0.032435
## Python:Visual Studio:Senior-C++:Visual Studio:Senior   -137.087893 -195.4930639 -78.682723 0.000000
```

Without drawing conclusions for all combinations presented by the Tukey test we can say that the highest combination for lines of code is **Language = C++ and Experience = Senior**.

b) How do it compare with your 'true' model you defined in the simulation

When we compare the created model defined in our main function with the defined coefficients, with the data generated by the model, we can see that they are similar. Also we can notice some variations that can be explained either by the defined standard deviation of the model, or that the model interpreted the data in another way than what we set up. The difference in the first order is very small compared to our true model, however the third order interactions are deviating quite a bit compared to our true model.

We can observe the differences looking in the following table and the defined factors:

	Defined Coeficients	Model Coeficients
(Intercept)	100	97.970697
LanguageC++	50	58.766665
LanguagePython	-30	-30.069280
IDEVisual Studio	-10	-3.958318
ExperienceSenior	30	40.490248
LanguageC++:IDEVisual Studio	20	-6.021773
LanguagePython:IDEVisual Studio	-20	-0.907659
LanguageC++:ExperienceSenior	10	-5.140028
LanguagePython:ExperienceSenior	10	15.543527
IDEVisual Studio:ExperienceSenior	20	10.129117
LanguageC++:IDEVisual Studio:ExperienceSenior	10	28.139782
LanguagePython:IDEVisual Studio:ExperienceSenior	-15	-45.909835

c) How do the results change compared to the correctly powered experiment.

When we compare the two models we see some differences between them. These could be explained by the standard deviation and the small sample size. We also get more coefficients with small p values that are to be considered statistically significant for m1.

Comparing the analysis on the two models show similar result for Hypothesis 2 and 3 but for model two we do not see an interaction effect for IDE. This means we can't reject the null hypothesis for hypothesis 1 that the means are equal for the different IDEs.