

| LAB 2: Multiple  regression, Non-linear  regression, overfitting,  and cross-validation |
| --- |

Due date : October 15, 2023, 11:59 PM

To access assignments, go to **file>make a copy**.

The lab will be in your own google drive, and you can modify it.

Your last name:

Your first name:

Your student ID:

Your last name:

Your first name:

Your student ID:

**Lab 2: What makes a board game successful**

Board games have become exponentially more popular (and better) over the past years. Beer and board games share a similar story: a few decades ago, we only had a few dozen popular board games (e.g., Monopoly, The Game of Life, Scrabble, Clue); similarly, we only had a few dozen popular beer brands (e.g., Budweiser, Heineken, Coors, Molson). But with the rise of the hipster culture, lots of things have changed—some for better and some for worse. To its credit, hipster culture has generated a renewed interest in “craft” and “artisan” objects. For instance, artisan beer has become extremely popular — and, as a result, most cities are now able to enjoy tons of small microbreweries that serve good and tasty beer. The same happened with board games: we are now able to enjoy thousands of alternative board games, some of which are super fun, creative, and unique.[[1]](#footnote-0)

But not all board games are created equally: some are dreadfully lame. In this lab, we will study board game attributes to see what drives board game ratings. To this end, I have downloaded game attributes from boardgamegeek.com (the IMDb of board games). Our data includes the following variables:

| * **Response Variable** |
| --- |

| * **avg\_rating:** The average rating of the board game. This variable ranges between 0 and 100, and comes from boardgamegeek.com. |
| --- |

| * **Predictors** |
| --- |

| * **ID**   + **name:** The name of the game   + **game\_id:** The unique identifier of the game * **Quantitative**   + **ranking:** The game’s rank   + **avg\_timeplay:** On average, how long does it take to play one game   + **min\_timeplay:** minimum required time to play the game   + **max\_timeplay:** maximum required time to play the game   + **year:** release year of the game   + **num\_votes:** number of people rating the game   + **min\_players:** minimum number of players that can play the game   + **max\_players:** maximum number of players that can play the game   + **age:** recommended minimum age to play the game   + **weight:** board game weight (in pounds) * **Categorical**    + **designer:** who designed the game   + **category:** board game category |
| --- |

# 1. Model Issues: Non-linearity (5 points - Lecture 4)

Run the following model:

| avg\_rating = bo+ b1(year)+ b2(avg\_timeplay)+b3(weight) |
| --- |

1. (2 points) We will assess (i) the linearity of each predictor and (ii) the linearity of the model as a whole. Use the residualPlots() function from the “car” package, and paste the residual plots (and the numerical linearity results) below.

**Residual plots**

| **Insert plots here** |
| --- |

**Numerical Non-Linearity results**

(table displayed that includes Tukey test)

| **Insert numerical results here** |
| --- |

1. (1 point) Based on the results above, which variables violate the linearity assumption? How did you reach this conclusion?

| **Your answer:** |
| --- |

C. (2 points) What does the Tukey test tell you about our model? Is linear regression a good fit for this data, overall? What statistical techniques do you think we would need to use to improve the statistical fit of this model?

| **Your answer:** |
| --- |

# 

# 

# 2. Model Issues: Heteroskedasticity (5 points - Lecture 4)

1. (1 point) What does heteroskedasticity mean? What is the effect of heteroskedasticity on a linear model?

| **Your answer:** |
| --- |

Run the following model:

| avg\_rating = bo+ b1(avg\_timeplay) |
| --- |

1. (0.5 point) Paste the regression output below.

|  |
| --- |

1. (0.5 point) Do a visual funnel test. Paste the plot below. In one sentence, tell me if you find visual evidence of heteroskedasticity in the model.

| **Your plot:**  **Insert plot here**  **Your answer:** |
| --- |

1. (1 point) Do a NCV test. Paste the results below. What can we conclude from this test?

| **Paste results here:**  **Insert a screenshot of your R-output here**  **Your conclusion about the NCVTest:** |
| --- |

1. (1 point) Run a heteroskedasticity-corrected model and paste the regression output below:

| **Insert a screenshot of your R-output here** |
| --- |

1. (1 point) In your new model, what happened to (i) the coefficients and (ii) the t-value and p-values of the predictor?

| **Your answer:** |
| --- |

# 3. Model Issues: Outliers (4 points - Lecture 4)

Run the following model:

| avg\_rating = bo+ b1(min\_players)+ b2(age)+b2(num\_votes) |
| --- |

1. (1 point) Do a Bonferroni test to formally identify the outliers. Paste the Bonferroni test results below:

| **Insert a screenshot of your R-output here** |
| --- |

1. (1 point) List the name of the games that are flagged as outliers by the Bonferroni test:

| **Your answer:** |
| --- |

1. (2 points) Remove any outliers you identified, and re-run the regression without these outliers. Did you notice any changes in the model?

| **Your answer:** |
| --- |

# 

# 

# 

# 4. Model Issues: Collinearity (6 points - Lecture 4)

1. (1 point) Define, in your own words, what collinearity is between two variables. Why is it problematic?

| **Your answer:** |
| --- |

Suppose you want to run the following model:

| avg\_rating = bo+ b1(year)+b2(age)+b3(min\_timeplay)+b4(max\_timeplay) |
| --- |

1. (1 point) Before running this model, you need to make sure that there is no collinearity between the predictors. Create the correlation matrix for the predictors.

| **Insert correlation matrix here** |
| --- |

1. (1 point) Based on this correlation matrix, which pair(s) of predictors are collinear?

| **Your answer:** |
| --- |

1. (1 point) Now, run a VIF test for the same model. Based on this test, which variables are collinear?

| **Your answer:** |
| --- |

1. (2 points) Why do you think there is collinearity between these two variables? Would you have suspected it? Why? Explain your reasoning in one short paragraph.

| **Your answer:** |
| --- |

1. (Optional) Using the VIF test as a reference, get rid of the collinearity problem. Re-run the model (you may eliminate any of the collinear variables). Do you notice any change for any of the coefficients?

| **Your answer:** |
| --- |

# 

# 5. Presentingprofessional regression tables (3 points)

1. (1.5 points) As a business data analyst, your responsibility is to not only design and run predictive models, but to also present these results to managers, policymakers, executives, and board directors.

For this reason, the goal of this course goes beyond simply running regressions. We will also learn how to interpret and present great statistical reports—reports that are beautiful, sleek, well-written, and aesthetically pleasing.

We will now learn how to present regression tables. If you simply copy/paste the regression output from R (as you’ve done so far), your reports will look unimpressive and messy. Fortunately, there is a tool called Stargazer, which is ubiquitously used by statisticians. Stargazer takes your R output and formats it in a manner that is internationally accepted by the data-science community.

To create a table using Stargazer, follow the steps below:

* **Step 1:** Open the board\_games dataset and run the following four linear regressions (call them *reg1*, *reg2*, *reg3*, *mreg*)

| * reg1: avg\_rating = bo+ b1(avg\_timeplay) * reg2: avg\_rating = bo+ b2(min\_players) * reg3: avg\_rating = bo+b3(max\_players) * mreg: avg\_rating = bo+ b1(avg\_timeplay)+ b2(min\_players)+b3(max\_players) |
| --- |

* **Step 2:** Run the following code:

| >install.packages(“stargazer”)  >library(stargazer)  >stargazer(reg1, reg2, reg3, mreg, type="html") |
| --- |

* Step 3: You should get a long html code. Copy this code, and paste it on this link below:

<https://www.w3schools.com/tryit/tryit.asp?filename=tryhtml_default>

(Note: after opening the website link, you’ll find some pre-written code. Delete this code, paste your own, and click *run*).

* **Step 4:** You should be getting a nice looking table with four regressions. Below, paste a screenshot of this table:

| **Insert your stargazer table screenshot here** |
| --- |

As you can see, we have four different regressions, which are compiled in a single table. This is the standard format for presenting tables in the data science community. Each column represents a different regression: Column (1) represents *reg1*, Column (2) represents *reg2*, etc.

1. (0.5 point) In R, type *summary(reg2)* and compare the R output with your stargazer table (in Column 2). Now, tell me, what does the value -0.138 represent in this regression, and what does the number below it (0.012) represent?

| **Your answer:** |
| --- |

1. (0.5 point) Now, I’d like you to present the tables with proper variable names. It’s certainly unprofessional to present the variables with the name “min\_players” or “max\_players.” Stargazer allows you to rename variables. I would like you to rename, within the stargazer code, the variable names to “Board game rating,” “Average timeplay,” “Minimum number of players,” “Maximum number of players.” **In other words**, I’m not asking you to rename the variables in the dataset, but rather to rename in the stargazer code (Hint: This [link](https://cran.r-project.org/web/packages/stargazer/vignettes/stargazer.pdf) might help you):

Paste your code:

| **Your code:** |
| --- |

Below, paste the stargazer table with proper variable names:

| **Insert your stargazer table screenshot here** |
| --- |

1. (0.5 point) Now, present the stargazer table with the coefficients and standard errors rounded to two decimal places (Hint: This [link](https://cran.r-project.org/web/packages/stargazer/stargazer.pdf) might help you):

Paste your code:

| **Your code:** |
| --- |

Below, paste the stargazer table with results using only two decimal places:

| **Insert your stargazer table screenshot here** |
| --- |

If you look at any scientific statistical report, you’ll find that scientists use this exact format. For example, look at page 10 of this [report](https://www.hbs.edu/faculty/Publication%20Files/19-123_ea5e9c88-8207-4aef-acb5-b206333b70dc.pdf) (from Harvard professors), or any report for that matter.

There you go—you’ve become a star….gazer!

# 6[. Polynomial regression:](https://www.youtube.com/watch?v=2jjeby9ZUR8&frags=pl%2Cwn) Age

# (5 points - Lecture 5)

Let’s figure out the true functional relationship between *avg\_rating* and *age*:

| avg\_rating = f(age) |
| --- |

1. (2 points) Try running four regressions: a linear (d=1), quadratic (d=2), cubic (d=3), and quartic (d=4). Paste the output of these four regressions below (using stargazer). This time, I want a table with five coefficient rows: one row for the intercept and one for each polynomial degree. Note: To achieve this, you should rename the coefficients in the table’s output so that all polynomials of the same degree are in the same row.
   * For example, **poly(age,3)2** and **poly(age,4)2** should be renamed as **age2** and appear in the same row.
   * **poly(age,3)3** and **poly(age,4)3** should be renamed to **age3** and appear in the same row.

| **Insert your stargazer table here** |
| --- |

B. (2 points) Create a matrix of four scatterplots (two columns, two rows) with the corresponding polynomial fit in each graph, **using GGPlot**. Make sure that the fitted polynomials are in **blue**, the dots are in grey, and that there is a legend (indicating the degree of the polynomial). Paste the matrix of graphs below. Hint: par(mfrow) doesn’t work with GGPlot, to create a matrix of plots. But you can use ggarrange(), following the commands from this [website](http://www.sthda.com/english/articles/24-ggpubr-publication-ready-plots/81-ggplot2-easy-way-to-mix-multiple-graphs-on-the-same-page/):

| **Insert your matrix of plots here (Using GGplot)** |
| --- |

C. (0.5 point) Run an ANOVA test to determine the optimal polynomial model (between the linear, quadratic, cubic and quartic regressions). Post the results below:

| **Insert a screenshot of your ANOVA output here** |
| --- |

D. (0.5 point) Based on the above results, which model would you consider appropriate to describe this functional relationship ?

i) Linear (d=1) \_\_\_\_\_\_

ii) Quadratic (d=2) \_\_\_\_\_\_

iii) Cubic (d=3) \_\_\_\_\_\_

iv) Quartic (d=4) \_\_\_\_\_\_

# 7. Polynomial regression: avg\_timeplay

# [(5 points - Lecture 5)](https://www.youtube.com/watch?v=9Fv5cuYZFC0&frags=pl%2Cwn)

Let’s figure out the true functional relationship between *rating* and *avg\_timeplay*:

| rating = f(avg\_timeplay) |
| --- |

1. (2 points) Try running four regressions: a linear (d=1), quadratic (d=2), cubic (d=3), and quartic (d=4). Paste the output of these four regressions below (using stargazer). This time, I want a table with five coefficient rows: one row for the intercept and one for each polynomial degree. Note: To achieve this, you should rename the coefficients in the table’s output so that all polynomials of the same degree are in the same row.
   * For example, **poly(x,3)2** and **poly(x,4)2** should be renamed as **x2** and appear in the same row.
   * **poly(x,3)3** and **poly(x,4)3** should be renamed to **x3** and appear in the same row .

| **Insert your stargazer table here** |
| --- |

1. (2 points) Create a matrix of four scatterplots (one column, four rows) with the corresponding polynomial fit in each graph, using GGPlot. Make sure that the fitted polynomials are in **green**, the dots are in grey, and that there is a legend (indicating the degree). Paste the matrix of graphs below:

| **Insert your matrix of plots here (Use GGplot)** |
| --- |

1. (0.5 point) Run an ANOVA test to determine the optimal polynomial model (between the linear, quadratic, cubic and quartic regressions). Post the results below:

| **Insert a screenshot of your ANOVA output here** |
| --- |

1. (0.5 point) Based on the above results, which model would you consider most appropriate to describe this functional relationship ?

i) Linear (d=1) \_\_\_\_\_\_

ii) Quadratic (d=2) \_\_\_\_\_\_

iii) Cubic (d=3) \_\_\_\_\_\_

iv) Quartic (d=4) \_\_\_\_\_\_

# 

# 8. Multiple polynomial regression

# (2 points - Lecture 5)

Consider the following relationship:

| avg\_rating = f(age, avg\_timeplay) |
| --- |

1. (1 point) Run the regression using the best polynomial relationship you found for *age* (in Question #6) and *avg\_timeplay* (in Question #7) , and combine them in a multiple polynomial regression. Please write the multiple polynomial regression equation to be estimated (in the form of **avg\_rating=*bo*+*b1*.....)**

| **Your regression equation:** |
| --- |

1. (1 point) Run the multiple polynomial regression, and paste the results of this regression below:

| **Insert your stargazer table screenshot here** |
| --- |

# 9. Spline regression (5 points - Lecture 5)

1. (1 point) What is the difference between a polynomial regression and spline regression? Why would we want to use splines as opposed to polynomials?

| **Your answer:** |
| --- |

1. (2 points) Let’s focus on the following relationship :

| avg\_rating = f(avg\_timeplay) |
| --- |

For the above relationship, run three spline regressions with three knots each: (i) a linear spline; (ii) a quadratic spline; and (iii) a cubic spline. Space the knots uniformly across the quantiles of the data, using the method we learned in class. Please paste the three regressions output below (using stargazer)

| **Insert your stargazer table screenshot here** |
| --- |

1. (2 points) Create a matrix of three distinct scatterplots, using GGPlot. In each graph, put each regression fitted splines, with (i) splines in red and (ii) vertical dashed lines where the knots are located; and (iii) a legend indicating the degree of the spline, and the number of knots.

| **Insert your matrix of plots here (Use GGPlot)** |
| --- |

# 10. Validation set test (4 points - Lecture 6)

1. (3 points) Consider the following statistical relationship:

| avg\_rating = f(weight) |
| --- |

Perform a validation set test for different polynomial models (d=1…,10). Repeat this test 30 times for each model, and paste the average of the 30 tests. You should write down the 10 average MSEs. Note: you need to automate these 30 tests using loops (write the code below):

| **Your R code:** |
| --- |

| Paste the average MSEs of the 30 tests for each model:   | **Average MSEs (3 decimal points):**  **d=1:**  **d=2:**  **d=3:**  **d=4:**  **d=5:**  **d=6:**  **d=7:**  **d=8:**  **d=9:**  **d=10:** | | --- | |
| --- | --- |

1. **(1 point) Based on your results, which polynomial function would you use?**

| **Your answer:** |
| --- |

# 11. LOOCV test (7 points - Lecture 6)

# **Suppose we want to test the following statistical relationship:**

| avg\_rating = f(weight) |
| --- |

**A (1 point) In a given trial of a LOOCV test (in our dataset), how many sets are created, and what are their sizes?**

| **Your answer:** |
| --- |

**B. (1 point) In our dataset, how many tests would a LOOCV perform?**

| **Your answer:** |
| --- |

**C. (1 points) Perform a LOOCV for the simple regression model (d=1). Put a timer before you run the test. What was the result of this test?**

| **Your answer:** |
| --- |

**D. (3 points) Now, run a LOOCV for the ten models (d=1,…10). Note: This test will take between a few minutes.**

| **Average MSEs (3 decimal points):**  **d=1:**  **d=2:**  **d=3:**  **d=4:**  **d=5:**  **d=6:**  **d=7:**  **d=8:**  **d=9:**  **d=10:** |
| --- |

**E. (1 points) Based on your results, which polynomial function would you use?**

| **Your answer:** |
| --- |

# 

# 12. K-fold cross-validation (5 points - Lecture 6)

1. (1 point) Suppose I have a dataset with 30 observations=(Y1,X1), (Y2,X2),…,(Y30,X30). If I have five folds, how many tests would I perform?

| **Your answer:** |
| --- |

We want to perform a K-fold cross validation for the statistical relationship:

| avg\_rating = f(num\_votes) |
| --- |

1. (3 points) Perform a 5-fold test for the ten models (d=1, d=2,..., d=10). Make sure the test is automated. Paste the code and the results below:

| **Your code:** |
| --- |

| **Average MSEs (3 decimal points):**  **d=1:**  **d=2:**  **d=3:**  **d=4:**  **d=5:**  **d=6:**  **d=7:**  **d=8:**  **d=9:**  **d=10:** |
| --- |

1. **(1 point) Based on your results, which polynomial function would you use?**

| **Your answer:** |
| --- |
|  |

# 13. Cross-validation in a multiple spline model

# (5 points - Lecture 6)

# Consider the following relationship:

| avg\_rating = f(age, year, num\_votes, avg\_timeplay) |
| --- |

We want to have the model with the best out-of-sample performance (aka. predictive performance). Each spline will have three knots, at the 25th, 50th, and 75th percentile of the distribution of each predictor:

*avg\_rating*=*fspline*(# of knots=3,degree=a)(*age)+ fspline*(# of knots=3,degree=b)*(year)*

*+ fspline*(# of knots=3,degree=c)*(num\_votes)+ fspline*(# of knots=3,degree=d)*(avg\_timeplay*)

We want to determine if we should use

i) a linear spline (degree=1)

ii) a quadratic spline (degree=2); or

iii) a cubic spline (degree=3); or

iv) a quartic spline (degree=4); or

v) a quintic spline (degree-5)

In other words, you will need to find the optimal combination of (a,b,c,d). Note, there are hundreds of combinations possible.

To find this optimal combination, you will run a K-fold validation set (with K=20). In other words, you will need a 20-fold test for each combination of (a,b,c,d), and then determine which combination gives you the lowest MSE. Hint: You will need to program multiple loops.

(Depending on how efficient your code is, the calculations should take between 5 to 30 minutes on a regular computer.)

A. (3 points) Paste your code below:

| **Your code:**  **Note: Two years ago, for a similar question, someone shared her/his code with a FRIEND, in good faith—this friend, in turn, shared it with a group of friends, and so on. Long story short, about 70% of the class ended up with the exact code (and you can imagine the rest of the story). While you’re encouraged to collaborate, do not pass along codes as a “favour” to your classmates—no matter how much you trust them—because these codes will probably spread out. I would like to avoid this situation from ever happening, by warning you ahead of time. This is not an easy question, but it’s a fantastic one to train to become a good coder. Just do your best and take it as a challenge to train your coding skills. I am generous when giving part marks for good attempts. But if anyone asks you for the code as a favour, just tell them: “*Stargazers never give up their codes*”** |
| --- |

B) (2 points) Which model has the best out-of-sample performance?

| **Your answer:**   * **a:** (1, 2, 3, 4, or 5?) * **b:** (1, 2, 3, 4, or 5?) * **c:** (1, 2, 3, 4, or 5?) * **d**: (1, 2, 3, 4, or 5?)   **Your MSE for this optimal combination:** |
| --- |

Note: This is how data scientists test models in real life. But they typically have super computers running this type of test for very large statistical models, looking for millions of possible combinations to continually improve their predictive capabilities! In fact, during the midterm you’ll have to predict a real life event, and you’ll be doing much of this to find the best prediction.

* **Instructions:** Please save **in colour** as a PDF and submit by the beginning of class, via mycourses. If you don’t submit as a PDF, you’ll be subject to a 2-point deduction
* Please attach your code when you submit Lab 2 (as a separate file). It will help us see where you made mistakes
* **Due date:** October 15 at 11:59 pm (Montreal time).

1. I personally recommend you play “Splendor.” Such a cool board game. [↑](#footnote-ref-0)