Christine Joan Emmanuela

Professor Koehler

Data Bootcamp

12 May 2025

Predicting a Broadway Show's Weekly Gross Revenue

I. Introduction

It goes without saying that Broadway is a large business, contributing to New York City's economy, providing jobs to numerous people all the way from creatives to analysts. Forbes once recorded that "overall, Broadway contributes $14.7 billion to the economy of New York City on top of ticket sales and supports 96,900 local jobs." (Sands 2023) In this project, I aim to build a model that will predict a Broadway show's gross revenue for the week. This model will be mainly useful for investors and producers who wish to predict the earnings of a show on a given week and get an idea of how the show is performing in the industry. This model will also be useful for deciding on the pricing of a ticket, as it will take into account the demand of consumers given a price level. The data used in this model is scraped from *Playbill's* website, sourced from *The Broadway League* (where all data on Broadway are held). The exploration and modeling process reveal that the most helpful features in predicting weekly gross revenue are average ticket price, number of seats in the theatre, number of awards, season, and genre. Having tried 4 different models, the KNN Regression model proves to be the best one.

II. Data Description

Broadway's main source of data can be found through *The Broadway League*. (The Broadway League 2025) However, the downloadable information is on a weekly basis, and in

order for the data to be useful, we need to have at least 3 months of data. For this project, I want to get a hold of 1 year worth of weekly gross data in order to catch seasonal trends. Yet because downloading 52 files of individual data would be too much, I decided to scrape it from the website directly instead. Originally, I planned to scrape directly from *The Broadway League's* website. However, because their dropdown menu is scripted in Java, my scraping code wasn't able to access the different weeks. Thus, after conducting a thorough search across Broadway affiliated websites, I found the same exact data displayed on *Playbill's* website, which they source from *The Broadway League*. (Playbill 2025) Furthermore, they have individual *url*s from each week that have a predictable format. I was then able to use the function I already made to scrape the data and just had to make a url list dating 52 weeks back, concatenating all the data into one dataframe. Because the code is scraping a lot of data, it takes a really long time for it to load. However, I concluded that scraping was still the better option compared to downloading individual files. Because the data scraping in itself takes more than 30 minutes, I decided to make it a separate colab notebook, which I then saved into a .csv file (uploaded unto GitHub). The .csv file will then be directly uploaded to the EDA and Modeling notebooks to avoid repeated data scraping.

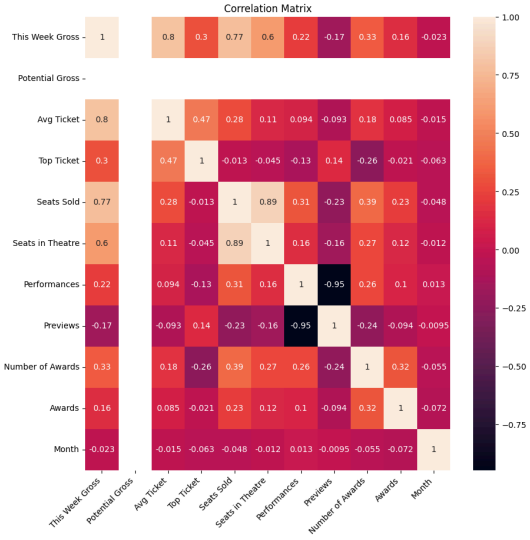
Now that I have the data scraped into one dataframe, I was able to do some feature engineering and add some features that I wanted to explore. The raw data scraped from the website consists of the Broadway shows, which I set as the index, and their Weekly Gross Revenue, Theater, Average Ticket Price, Top Ticket Price, Seats Sold, Seats in Theater, and % Capacity of seats filled during the week. While scraping the data, I decided to scrape deeper into the embedded links to access the individual show information and scrape their genre and the number of awards they had. Furthermore, I created a binary column which will indicate whether

or not a show has an award. It appears that the excessive and deep data scraping is the key reason behind why it takes so long to scrape. After scraping, I feature engineered further by adding a season and month column since I want to capture the seasonality of Broadway shows and see how they vary from season to season. With a total of 1661 rows worth of one year data and 19 columns of both categorical and numeric data, I am now ready to conduct data exploration.

III. Exploratory Data Analysis

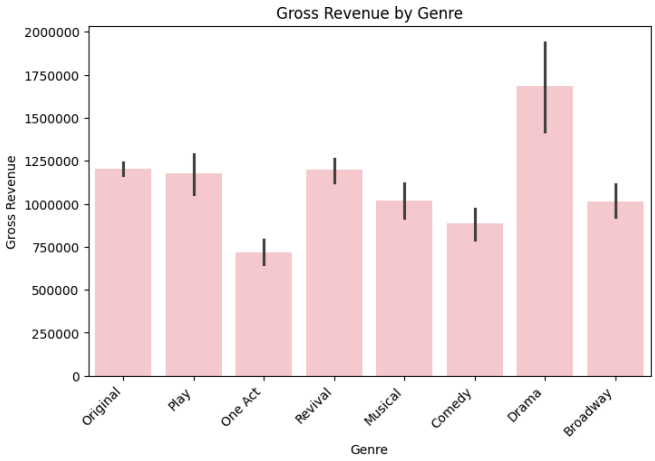
The data exploration is conducted with the aim to see which features would be most helpful in building the model that will predict weekly gross revenue. I started with a heatmap to see the general correlation between the features and then furthered on to individual features vs. Weekly Gross Revenue. I end my analysis of each graph with a note of whether or not they should be used in the model.

1. *Which features are most correlated with gross revenue?*



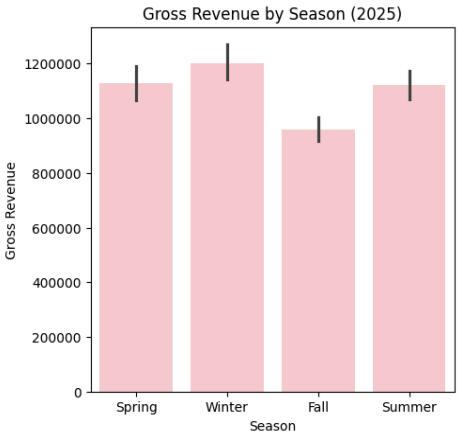
Average Ticket Price seems to have the highest correlation with Gross Revenue with a correlation of .8, followed by Seats Sold with a correlation of .77. Both these features feel intuitive, but that's because multiplying average ticket price with the number of seats sold would easily show the gross revenue. Therefore, for my model it's important to keep in mind that I should just use one of these features. Other important features are: Seats in Theatre (.6), Number of Awards (.33), Top Ticket Price (.3)

2. *How does gross revenue vary by genre?*



Dramas and Revivals have the highest gross revenue, and one act and comedy have the least. The gap is significant with a difference of around $1,000,000 (Drama-One Act). Genre would be worth using in the model.

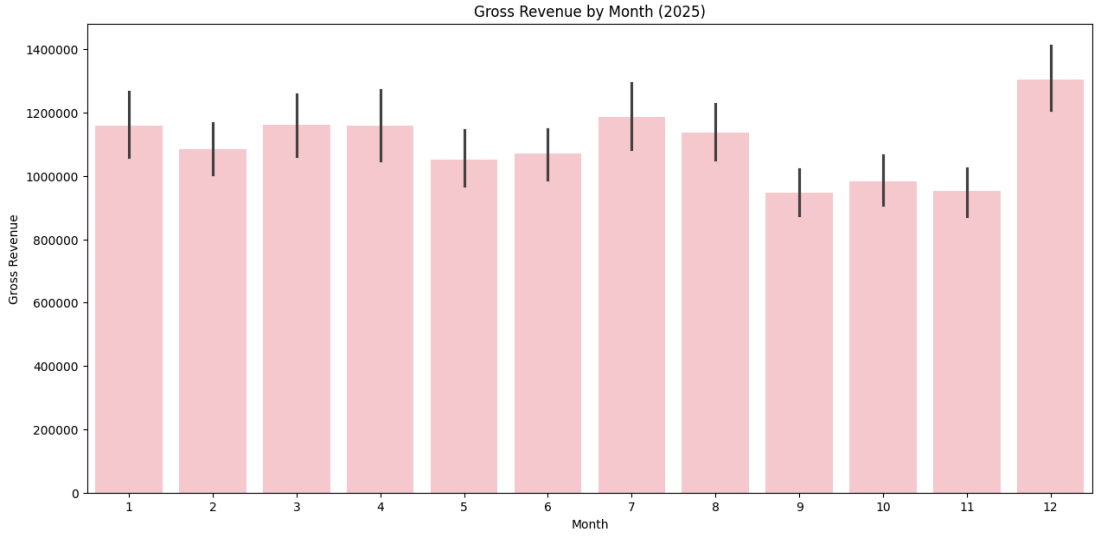
3. *Does seasonality impact gross revenue?*



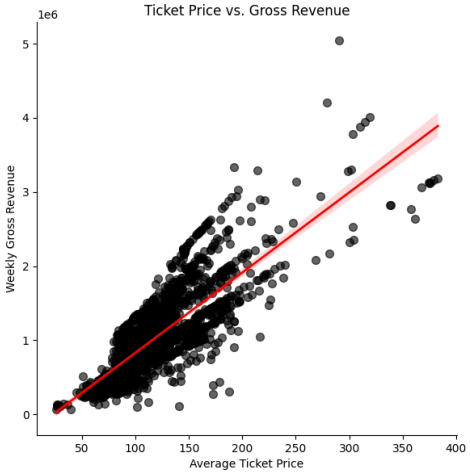
Yes. There is higher gross revenue during Winter and Summer, which are peak tourist seasons. Season would be worth using in the model.

4. *Does Month impact gross revenue?*

Yes, December (Peak Winter Tourist Season) brings the highest gross of around $1.3 million, with July (Peak Summer Tourist Season) following up with a gross of around $1.19 million. After the peak of summer, September brings the lowest gross of $950k. The gap of $350k between the months shows that month is worth using in the model.

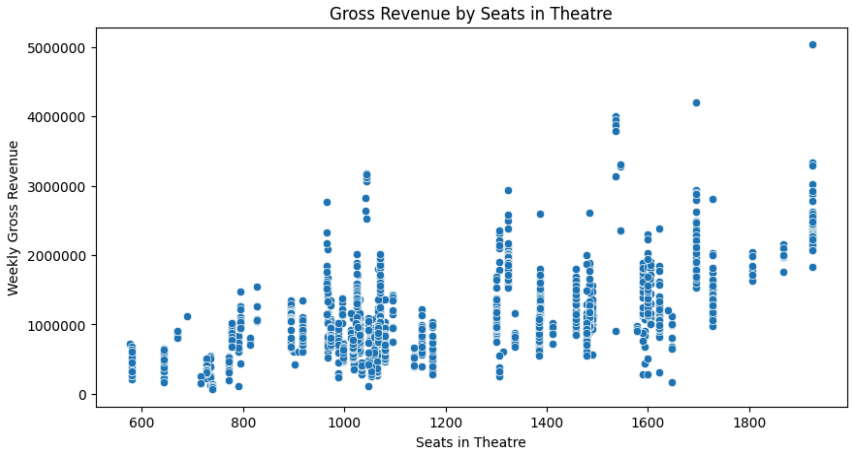


5. *How does average ticket price relate to gross revenue?*



There appears to be a strong positive correlation between ticket price and gross revenue. However, once ticket price becomes higher than USD 200, there is no longer a strong correlation, implying that audience's demand generally stops at USD 200. That being said, below USD 200, Avg Ticket is a strong predictor of Weekly Gross Revenue. This should definitely be a part of the model.

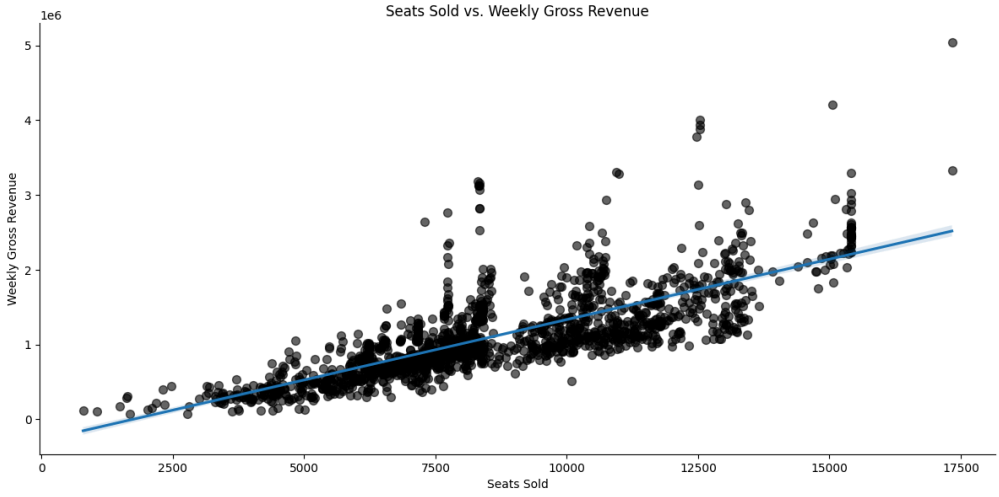
6. *Do theaters with more seats make more money?*



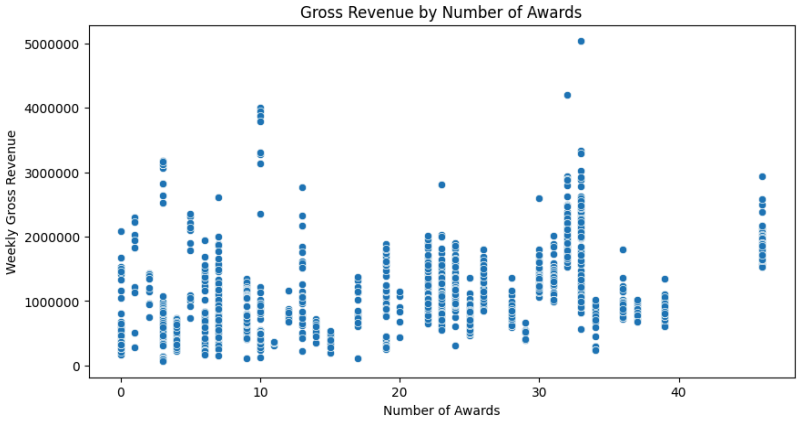
Yes, there is a slight positive correlation between seats in Theatre and Weekly Gross Revenue. Nothing too dramatic, but consistent. It's worth trying to use this feature.

7. *Is Seats sold a good predictor of gross revenue?*

Yes, there appears to be a strong positive correlation between the number of seats sold with the weekly gross revenue. This is a solid predictor (and intuitive), but should not be used together with average ticket price.



8. *Is there a correlation between the number of awards a show has and its gross?*



No, there is no significant correlation between the number of awards a show has and its gross. This feature shouldn't be added to the model since it would be unhelpful.

In conclusion, the features that I should start building the model with are: Average ticket price OR seats sold (pick one), genre, seasonality, top ticket price, month, and seats in theater.

IV. Models and Methods

Having explored the dataset, we will now start the model-building part of the project. The aim is to predict weekly gross revenue. This project will build 4 different models and see which one performs best. Because this project deals with Gross Revenue, we will be using Regression models. For each model, I will use a 80-20 train-test split, training my model on 80% of the data and testing it on 20%. To evaluate the models, I will be using the Root Mean Squared Error and Coefficient of Determination (R^2) of both the test and train data and compare it against the baseline and other models. With the context of this problem, I chose to use Root MSE because it's more applicable to the situation of predicting Weekly Gross Revenue. For example, the baseline has an MSE of 334670160593.26263 but a RMSE of 578506.837118856. In the context of this problem, a standard error of $578,507 makes more sense compared to $334,670,160,593

1. Baseline

For my baseline I took the mean of the weekly gross revenue and took the mean squared error. The results were as follows:

MSE: 334670160593.26263

RMSE: 578506.837118856

2. Linear Regression

For my first model I chose to use linear regression because of the sheer nature of what I'm predicting: gross revenue. Since this is a numbers heavy regression problem, it made sense for me to start with linear regression. I like to treat this as my "second baseline" to compare with more complex models.

Test MSE: 25006854209.154354

Test RMSE: 158135.5564354657

Train RMSE: 182632.3040452306

Train R^2: 0.8997500753267593

Test R^2: 0.926978935403707

Already this model is doing better than baseline with a much lower test RMSE (72.6% decrease). The model is not overfit because train RMSE is still higher than test. Test RMSE shows that the model's predictions are off by about $158,135.5 which is normal given the weekly gross is in the millions. This serves as an excellent second baseline, and we can now see if the more complex models will be better than this, meaning they will be even better models to an already excellent model. From the features importance ranking, this model conveys that the average ticket price and number of seats in the theatre are the most important ones in building the model.

3. Random Forest Regression

I chose Random Forest Regression as my second model because this model doesn't assume a linear relationship, which will be good to compare against linear regression. This model is also helpful for the more complex relationships within this data.

Test RMSE: 189213.69897661227

Train RMSE: 80867.61317390707

Train R^2: 0.9803447709694341

Test R^2: 0.8954571705672739

This model is way too overfit, with a much higher test RMSE compared to train RMSE. From the feature importance ranking, this model conveys that average ticket price and number of awards are the most determining factors.

4. KNN Regression

The next model I chose to try is the KNN Regression, which bases its predictions off the nearest values. This model is simple and versatile, helpful in uncovering local trends to predict Weekly Gross. Because the prediction is focused on near neighbors, it shouldn't be affected by extreme values.

RMSE Test: 127492.02854343074

RMSE Train: 131726.2172386768

Train R^2: 0.9478477200104081

Test R^2: 0.9525370026723889

This is my best model so far. The test RMSE is lower than linear regression, but not higher than train RMSE, which means that this model is not overfit. Based on the feature list, the most influential features are average ticket price, number of seats in the theatre, and number of awards. This comes as a surprise considering we didn't see too much of a significant correlation

between number of awards and weekly gross during data exploration. Nevertheless, this model remains my best model.

5. Decision Tree

At this point, I am content with my KNN model. However, I wanted to try one more model–Decision Tree–which is known to be as versatile as KNN. Decision Tree is also known to be better with different data types, so I had to try this model before concluding that KNN is the best one.

```
RMSE Test: 172589.71389696642
RMSE Train: 8473.43493899664
Train R^2: 0.9997842017804796
Test R^2: 0.9130200976344741
```

There is severe overfitting in this data. The difference of 164116 between Test and Train RMSE is just too much. Thus, this model should be left behind. From the feature list, the most influential features are average ticket price and number of seats sold in the theatre, echoing what was conveyed by the other models.

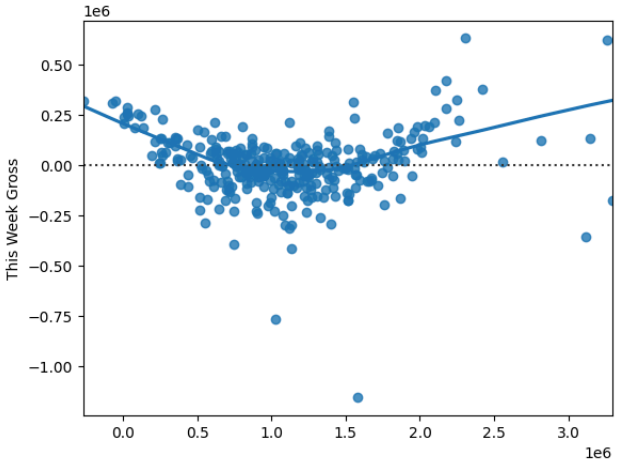
V. Results and Interpretation

Results DataFrame Comparing Different Models

	Model	Train RMSE	Test RMSE	Train R2	Test R2	Overfit?
0	Linear Regression	182632.304045	158135.556435	0.899750	0.926979	False
1	Random Forest Regression	80867.613174	189213.698977	0.980345	0.895457	True
2	KNN	131726.217239	127492.028543	0.947848	0.952537	False
3	Decision Tree	69336.521239	128396.700859	0.985550	0.951861	True

Overall, my KNN Regression model performed the best with a test RMSE of 127492 compared to baseline RMSE of 578506. This implies that my KNN Regression model is able to predict a show's Weekly Gross Revenue with a standard error of $127,492 which is great considering the large weekly gross average of $1,101,524 (y.mean()). The most impactful features were the average ticket price and the number of seats in the theatre, which is intuitive but important because these are both features that can be controlled. On the contrary, the number of seats sold is something beyond the producers' full control, which is why I left it out in the first place. Other influential features are the number of awards, genre, and season.

KNN worked well for this dataset because unlike linear regression which assumes linearity, KNN works with potential nonlinearity. In the residual plot from the linear regression model, we can see that there is a nonlinear pattern which signals that this dataset might be better off with a model that isn't linear. Because this model included categorical variables like genre and season, it made sense to use a model that would support this nonlinearity, which makes KNN a strong choice.



Another reason that a KNN model works well for this dataset is its nature that predicts based on similar/neighboring points. Because Broadway's gross revenue varies by season, as we established during data exploration, this KNN model seems to catch that seasonality and predicts based on that context.

VI. Conclusion and Next Steps

Ultimately, the KNN Regression model works best in predicting a Broadway show's weekly gross revenue without overfitting, and the most influential features are the average ticket price and the capacity of the theater. To enhance the complexity and predictive ability of my model, there are several things I would like to do:

1. *Consider Star Power in Predicting Weekly Gross Revenue*

   Lately, producers and directors have been putting in a lot of effort to get audiences back into the theatre in a post covid world. One way they have been doing this is by bringing in big star names. For instance, Robert Downey Junior, Idina Menzel, and Jeremy Jordan were all just recently or are currently still starring. This is one factor that I wish to include in my model next. However, because Broadway stars are constantly changing from run to run, this will be a very difficult but equally rewarding task. Including this feature would mean having to do a lot more scraping on the *Playbill* website to ensure the most up to date data.

2. *Scrape Review and Critic Scores to Enhance my Model*

   To get real time perspective on a show's performance, especially if the show is relatively new, audience and critic reviews would be most helpful. What I would like to do to further enhance the model is scrape audience and critic reviews online and use sentiment analysis and classification models to predict weekly gross based on reviews alone. If the model works, it would be very interesting and useful especially for new shows.

3. *Deploy the Model*

   Once I've enhanced my model with more categorical features, I would like it to reach a wider intended audience, and this means deploying my model into Streamlit. This way, potential investors, producers, or the general public are able to interact with my model and predict the weekly gross of a show.

Ultimately, with a working KNN model and room for enhancement, eventually deploying the model would be very beneficial for Broadway investors and producers. Investors are able to assess the potential of a show, and producers will be able to make production decisions (how much to price tickets, which stars to bring in) accordingly. Through data driven strategies, Broadway leaders are able to align creative ambitions with financial sustainability, seeing a world where Broadway continues to inspire the ton.

References

Playbill. 2025. “Grosses List.” Playbill. 2025. https://playbill.com/grosses.

Sands, Roger. 2023. “Broadway: The Engine That Helps Fuel New York City’s Economy.” Forbes. January 20, 2023. https://www.forbes.com/sites/rogersands/2023/01/20/broadway-the-engine-that-helps-fuel-new-york-citys-economy/.

The Broadway League. 2025. “Grosses - Broadway in NYC | the Broadway League.” Www.broadwayleague.com. 2025. https://www.broadwayleague.com/research/grosses-broadway-nyc/.