Word Target: 1,875 – 2,250

Taking the Guesswork out of Fantasy Baseball with Python for Data Analysis

# Introduction

Name, first career, grad school, today

# Where is this going?

* History of DFS and baseball
* Frame the problem by showing the variation in the population
* Rolling Mean
* Simple Exponential Smoothing
* Kernel Density Estimation (non-parametric)
* Beta – Binomial conjugate pair (Bayesion)
* Random Forrest (Modeling)
* Ensemble?
* Flask App
* Conclusion

# What is Baseball

Is it a child’s game?



Or is it art???





# Review what is DFS

<https://www.forbes.com/sites/darrenheitner/2015/09/20/an-abbreviated-history-of-fanduel-and-draftkings/>

<https://www.washingtonpost.com/news/wonk/wp/2015/07/28/how-daily-fantasy-sites-became-pro-sports-newest-addiction-machine/?noredirect=on&utm_term=.ac7ecd8e03bb>

< insert Fanduel and DraftKings logos >

# Review data to be used and Population Variance

< Data >

< Histogram >

< Box Plot >

# Rolling Mean

<https://towardsdatascience.com/implementing-moving-averages-in-python-1ad28e636f9d>

Reference: Forecasting: Principles and Practice. Rob J Hyndman and George Athanasopoulos

<https://otexts.com/fpp2/>

When making a forecast it is generally a good idea to use a simple method as a benchmark to compare other techniques to. In this instance, we will be using a rolling average



< Go into python code >

Go over results

# Simple Exponential Smoothing

Reference :

GILOVICH, T., VALLONE, R. and TVERSKY, A. (1985), “The Hot Hand in Basketball: On the Misperception of Random Sequences”, Cognitive Psychology, 17, 295–314.

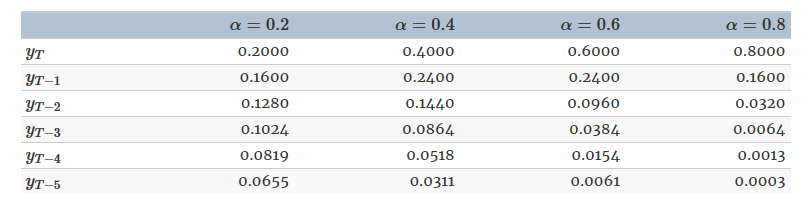
<http://www.cs.colorado.edu/~mozer/Teaching/syllabi/7782/readings/gilovich%20vallone%20tversky.pdf>

I chose this model to try to see if we could limit the variance by attempting to model when a player becomes hot. Although, the myth of the hot hand has been…

The hot hand phallacy or hot hand phenomenon is a belief that an individual who has success in the past will continue to have success continuing into the future. While the idea of the hot hand has been disproved in basketball is it possible that it exists in baseball with DFS?

In order to model this an exponential smoothing model was chosen. Simple exponential smoothing is a univariate time series forecasting method. It requires a single parameter, alpha (α), which controls the smoothing factor. This parameter takes values between zero and one. If alpha equals zero then the model is the average of the historical data. This means that equal weight is placed on each observation. If alpha equals 1, then this is a naïve method where the forecast is equal to the last value. If alpha is between 0 and 1, then a larger value means more weight is place on the latest observation and the least amount of weight is placed on the oldest value.

https://miro.medium.com/max/638/1*rWt9KIT0jVJzlIJ2sI8-Dg.png



< show python code >

First an instance of the Simple Exponential Smoothing class must be instantiated and passed the training data. The fit() function is then called providing the fit configuration specifically the alpha value called smoothing\_level. If this is not provided or set to None, the model will automatically optimize the value.

The forecast() or the prediction() function on the result object can be called to make a forecast. In our case we are making a one step ahead forecast, so the value is set to 1.

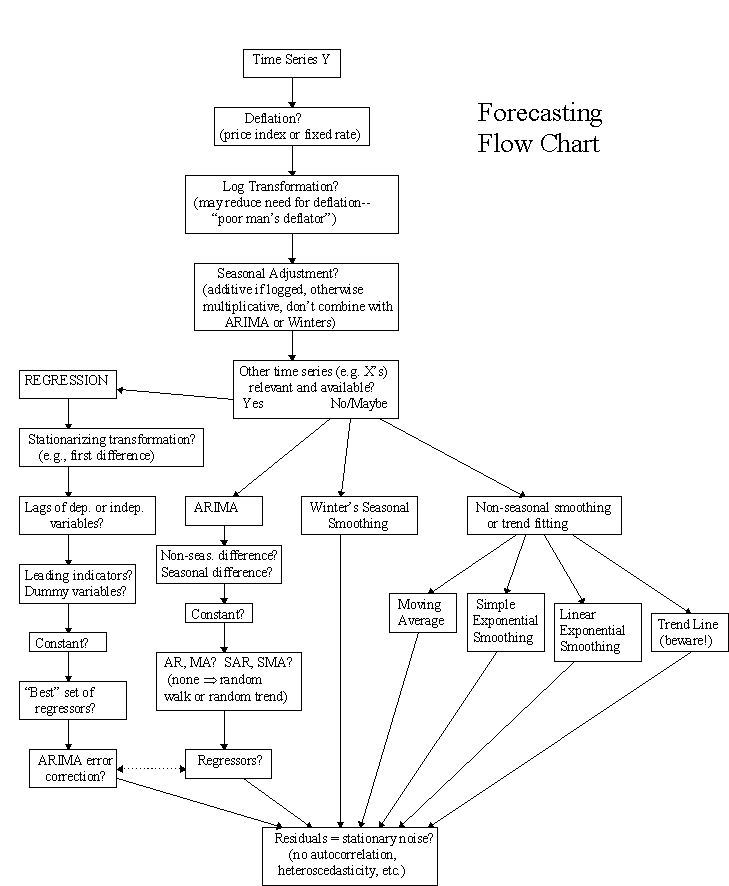
You may notice that we did not difference the data, well that is because there was not a strong trend and/or random-walk behavior.

Random walk behavior predicts that “next period equals this period” perhaps plus a constant.

Simple Exponential Smoothing – when data are nonseasonal and display a time-varying mean without a consistent trend. NO CLEAR TREND OR SEASONAL PATTERN

< results >

< results compared to rolling average >



<https://people.duke.edu/~rnau/411flow.gif>

# Beta – Binomial Bayesian Analysis

Reference: <https://mc-stan.org/users/documentation/case-studies/pool-binary-trials.html>

<https://towardsdatascience.com/bayesball-bayesian-analysis-of-batting-average-102e0390c0e4>

Partial pooling where the similarity among the items is estimated.

Baseball batting ability with y\_n hits in K\_n at bats for a baseball player.

With *complete pooling*, each item is assumed to have the same chance of success. With *no pooling*, each item is assumed to have a completely unrelated chance of success. With *partial pooling*, each item is assumed to have a different chance of success, but the data for all of the observed items informs the estimates for each item.

Partial pooling is typically accomplished through hierarchical models. Hierarchical models directly model the population of items. The population mean and variance is important, but the two hierarchical models we consider (chance of success vs. log odds of success) provide rather differently shaped posteriors.

From a population model perspective, no pooling corresponds to infinite population variance, whereas complete pooling corresponds to zero population variance.

* Review the pareto hyperparameter
* Why beta binomial
* Review the data
* Review the code for pyStan
* Brief overiew of pyStan
* Divergent Transitions and the mitigation strategies
* Show results for divergent and after the mitigation
* Review the results

Take the probability of a hit and multiple by point values to get expected points.

< Expectation Formula >

< Python code >

< Results >

< Compare Results >

# Kernel Density Estimation

Excerpt: <https://jakevdp.github.io/PythonDataScienceHandbook/05.13-kernel-density-estimation.html>

[*Python Data Science Handbook*](http://shop.oreilly.com/product/0636920034919.do) by Jake VanderPlas pages 491 - 505

What if we don’t know what is the likelihood? Or if the parametric form was not known?

Non-parametric density estimation to the rescue!

<http://research.cs.tamu.edu/prism/lectures/pr/pr_l7.pdf>

# Random Forrest

# Web App with Flask

# BONUS – General Additive Model on Strike Zone

# Conclusion

## Presentation:

<https://medium.com/learning-machine-learning/present-your-data-science-projects-with-jupyter-slides-75f20735eb0f>

<https://blog.kdheepak.com/jupyter-notebook-reveal-js-and-github-pages.html>

<https://medium.com/@mjspeck/presenting-code-using-jupyter-notebook-slides-a8a3c3b59d67>

<https://github.com/damianavila/RISE>

<https://nbconvert.readthedocs.io/en/stable/>