**Project Proposal**

**As of April 30, 2018**

All of my projects will use GitHub as a wrapper and will be presented as slides.

**Project 1: Slow Moving Inventory**

Slow moving inventory is hard to forecast and is traditionally modeled using the poisson distribution. Depending on the cost of the merchandise and the company inventory holding costs, it can be really expensive to stock inventory and even harder to offload. What if we could do a better job forecasting using another parameter or modeling the data with a nonparametric learner?

<https://www.kaggle.com/flenderson/sales-analysis>

**Dataset:**

* It has about 198K items. Not a lot of dimensions and not much cleaning to do.
* There are two variables that can be used as targets: binary (sold in the past 6 mos or not) for classification and SoldCount (how many items were sold) for regression.
* The x variables are limited: there’s MarketingType (how product was marketed), whether it’s a new product or not, release year (which is not really a knob we can turn, but it may be worth exploring), 3 price categories (PriceReg, LowUserPrice, LowNetPrice – I am thinking they are regular price, discounted price, and net cost of the product) plus a thing called StrengthFactor (whatever that is – it’s a very large number).
* There’s also an ItemCount, which I think is accumulated unsold inventory. So you could calculate how much the accumulated inventory is costing the company now and how much each percent improvement in forecasting would mean.
* Also there’s an order number and SKU number, so products can be aggregated by SKU.

**Next Steps:**

* EDA – see if I can generate some pretty charts to liven up the slides
* Determining which models to use and prepping the data accordingly (scale/standardize, treat null values, binary variables, categorical variables, etc.)

**Project 2: Converting Churned Customers to Active with Channel Profit Maximization**

This project is about ranking churned customers for promotional efforts designed to turn them back into active customers. Ranking would be based on the probability of churn. In order to have a train and test data set, I would have to split the existing dataset (which is rather large). Then using the different classification models (or different cutoffs for ranking scores), generate the matrix of expected rates, and the cost-benefit matrix, multiply, then plot the profit curves. (In order to go the second step, I would have to use fictional costs and profit figures.)

Since most companies have budgetary constraints, I would just look at what classifier would perform best at a certain budget constraint level. For a little novelty, I could plot profit curves for the whole channel (wholesaler plus retailer) then see if a channel coordinating profit proposal can be put together. E.g. if the wholesaler puts in an extra $x amount into targeting churned customers, an additional $y amount of channel profit could be generated and split between the wholesaler and retailer, enabling both companies to come out of the promotional effort with higher profit than without channel coordination.

Unless MIT comes through for me – I emailed them about the amazing case study we did two semesters ago and asked if they had a dataset to go with it. It is about forecasting meal kits for a fictional meal kit delivery company called Chef Yourself. It is exactly what I would like to do…

[http://archive.ics.uci.edu/ml/datasets/Online+Retail#](http://archive.ics.uci.edu/ml/datasets/Online+Retail)

**Dataset:**

* It has over 541K items. Downside is that it doesn’t have a lot of dimensions (8) but at least there’s some cleaning to do (there are minus quantities and prices!).
* The target variable is based on invoice date. It could be changed and tried with a few different approaches – do we consider people churned if they haven’t ordered in a month or in 6 weeks?
* For x variables, there’s country, customer id, unit price, and quantities ordered. I’m thinking country and customer id may be collated so there’s really only 3 variables to predict on. Dimensionality reduction will not be on the table.
* I could try to extract some useful data from the description. There are colors and there are some other elements (nouns like “bottle” “hanger” “lantern”) which I don’t think would be very informative.

**Project 3: Forecasting Beer Demand and Optimizing Production Planning**

This project comes from an amazing dataset on Kaggle on a fictional beer wholesaler. You are basically predicting two things: first, beer demand for January 2018 based on past demand using promotional data, pricing, weather data, soda sales, sporting events and other beer-worthy occasions, and customer demographics. Second, you need to make beer product-mix recommendations for two new retailers that are joining your company as clients. So that would be a really interesting recommender problem, where instead of people and movies, one’s using companies and beer types to make recommendations (reducing the beers to latent beer archetypes based on order-votes).

Obviously, this is great for regression but also for time series analysis (which has lots of awesome algorithms for smoothing and trends and seasonality and whatnot). Also, as a second—prescriptive analytics—step, one could forecast for the whole year and use fictional numbers to create either an optimized master production plan or optimize the distribution network now that there are two new customers.

<https://www.kaggle.com/utathya/future-volume-prediction>

**Dataset:**

* It has several csv’s (see list below). The data has lots of features but only 21K instances. There’s absolutely no cleaning involved.
* price\_sales\_promotion.csv: ($/hectoliter) Holds the price, sales & promotion in dollar value per hectoliter at Agency-SKU-month level
* historical\_volume.csv: (hectoliters) Holds sales data at Agency-SKU-month level from Jan 2013 to Dec 2017
* weather.csv: (Degree Celsius) Holds average maximum temperature at Agency-month level
* industry\_soda\_sales.csv: (hectoliters) Holds industry level soda sales
* event\_calendar.csv: Holds event details (sports, carnivals, etc.)
* industry\_volume.csv: (hectoliters) Holds industry actual beer volume
* demographics.csv: Holds demographic details (Yearly income in $)

**Next Steps:**

* Joining up the relevant columns from the separate csv’s then EDA.
* Going back to reviewing the many time series algorithms we used in my class and figuring out which models to use. Rebuilding the algorithms in python (or finding them in sklearn).
* Prepping the data (scale/standardize, treat null values, binary variables, categorical variables, etc.)

**Project 4: Analyzing the Effect of Promotions on Demand**

I do not have the dataset yet but my friend is giving me 3 years’ worth of wholesale sales data (2015-2017) for 100s of products as a csv. Csv will contain item number, description (messy string), unit price, sales in units, retailer, promotional info.

Sales of this product are extremely seasonal (45-48 % sales occur in the last quarter). Lead times are very long due to ocean shipping (the products are coming from Germany) so there is no time to adjust to changes in sales demand. Better forecasting is critical. Inventory falls into three main categories: slow-moving inventory (over 12 months) comprises one-third of products, less than 5 months on hand comprises another third, and the rest is in-between. Obviously, reducing inventory that sits there upwards of a year would be really beneficial.

The company is interested in determining how sales promotions are impacting sales – both the number of promoted items sold as well as the items that are not promoted (theory being that the promotions may be cannibalizing the sales of other fully-priced items.). Their idea is just vanilla regression – looking at sales of promoted products correlating (or not) with sales of unpromoted/undiscounted products. As well as looking at how sales promotions impact demand for promoted items – see more on this below.

My thinking was a bit different, because I think it would be beneficial to separate two different impacts on the data, seeking answers to two questions:

1. is demand driven by some underlying cause? (=regression question) Such as geography? Type of retailer? Or product category (see note below about data munging)? …or…
2. is there some overall pattern underneath the demand? (=time series question) Is there some sort of seasonality or trend in the data? (we already know that most definitely there is—it would be nice to separate the seasonality from other causal drivers.)

Another question I had is whether it would be best to segment products and analyze the different product categories separately. (Since we are using time series analysis to find underlying pattern in the data, we may see clearer patterns in aggregated product groups). One way to do this would be traditional ABC segmentation (segmenting products based on how much they contribute to the overall revenue). Another approach would be to generate baseline product categories by clustering unlabeled data. Let’s see if the predictions improve if we treat the clusters as separate categories and forecast on them differently.

About the sales promotions impacting sales. Not quite sure what the best baseline is for such a comparison. Should I pick a forecasting method that fits data without promotions best, and then create a column in the dataset for expected values without promotions and compare those to the values that materialized with promotions?.

And yes this would totally be done in two weeks. Oh sure.

**Next steps:**

* Get data!!!
* Munge it. Extract value from messy product description strings – products repackaged in US have English descriptions, products made in Germany go by German name. if it could be somehow standardized, the product descriptions could yield product categories – e.g. 7 out of 25 slowest products are Kochmesser of different types. See if any of the item numbers could be fixed. There seem to be items with 4-digit codes, items with four-digit codes plus slash and another two-digits, and items with four-digit codes plus slash plus lots more data filled in. Need to make sure that these are not the same items.
* EDA
* Clustering.