Code Test Responses

Brendan Herger, 13herger@gmail.com

Nota Bene: This report is aimed solely at a non-technical audience. For a technical discussion, please refer to the adjoining code, discussed in `README.md` in the original GitHub repository. Original questions, as provided, are in `docs/codetest_instructions.txt` .

Part 1: Model building on a synthetic dataset

Overview

Q) (Summary) Given a synthetic data set, create an algorithm capable of predicitng a real-valued response. Then create a set of predictions for the test data set.

A) I've placed an emphasis on creating a quick to implement, easy to modify and easy to extend framework, largely at the cost of a better mean squared error and scalability. Given the limitations of this test, I'm happy with this framework as a first pass, but would work with the product owner to understand the priorities for a second pass.

Enabled by the small data set size, I've elected to build my response using Pandas & SKLearn. In particular, I've used SKLearn's Pipeline & Grid Search modules to quickly implement a pipeline with the following components:

- Imputer: I've used SKLearn's Imputer to fill in missing values, which occur in a non-regular pattern through the dataset.
- PCA: Due to the high dimensionality of the dataset, I've used Principle Component Analysis (PCA) to reduce the dimensionality of the dataset
- Algorithms: I've used Ordinary Least Squares (OLS), Random Forest (RF), and Gradient Boosting Machine (GBM) separately to build predictive models.

With the components above, I've build a grid search with a moderately large parameter space for each step. This grid is then trained on a cross validation data set and ranked by average mean square error to arrive at the best predictive model.

Predictions

The predictions were created with the best ranked estimator, which happened to be an OLS based pipeline. The predictions are in a file called `part1_test_predictions.csv` , and the pickled SKLearn estimator is in a file called `best_estimator.pkl`

Future Work

If a more powerful predictive model were necessary, I'd propose looking at the following areas for future work:

- Ensemble model: A natural progression would be to ensemble the best performing variation of the three algorithms evaluated, perhaps with OLS or an averaging scheme
- More algorithms: It could be also interesting to evaluate a neural network, ridge / lasso regression, and / or nearest-neighbor approaches. Given a deeper understanding of the dataset, it would be possible to make a more informed choice, or without a deeper understanding it would be simple to include these methods in the current grid search.

- Include more data types: Currently, only numeric types are included in modeling, though there are a few non-numeric types in the dataset.

Part 2: Baby Names!

Overview

This section responds to a series of questions, centered around analyzing baby first names from a dataset provided by the US Social Security Administration.

As an aside, I've assumed that all normalization has occurred upstream. Frankie might be similar to Franklin and Franky, but for the purposes of this analysis I've treated them as three separate names.

Section A: Descriptive Analysis

Descriptive Analysis Q1

Q1) Please describe the format of the data files. Can you identify any limitations or distortions of the data?

A1) The data is contained in a zipped directory. Within the zipped directory is a directory containing multiple `.TXT` files, as well as a `README`. The `.TXT` files appear to be comma delimited files. It would appear that the data lacks state-level responses when a name has less than 5 occurrences for a state-year combination, and that DC is treated as a state. Otherwise, it looks to be a rather well-behaved data set.

Descriptive Analysis Q2

Q2) What is the most popular name of all time? (Of either gender.)

A2) For the US, for the data provided, the most common name is James, with 4,972,245 occurrences across 106 years.

Descriptive Analysis Q3

Q3) What is the most gender ambiguous name in 2013? 1945?

A3) The phrasing of this question is also somewhat ambiguous, so I've made a few assumptions

- Gender ambiguous means that there are as many boys as there are girls with that name
- Edge cases are not interesting. If only 10 people in the US share a name, it might be ambiguous but not popular enough to take the award.

  With this in mind, I've produced the following results for 2013 and 1945, with various minimum popularity cutoffs. Where ties occurred (such as with no minimum number of observations) the 'winner' is chosen arbitrarily:

birth_year	min_num_observations	most_neutral_name
2013	0	Nikita
2013	10	Nikita
2013	100	Jael
2013	1000	Charlie
2013	10000	Avery
1945	0	Maxie
1945	10	Maxie
1945	100	Lavern

1945	1000	Frankie
1945	10000	Jerry

Descriptive Analysis Q4

Q4) Of the names represented in the data, find the name that has had the largest percentage increase in popularity since 1980. Largest decrease?

A4) Similar to q3, I've made some assumptions:

- Edge cases are not interesting. If only 10 people in the US in 2013, it might have a large change but not popular enough to take the award.
- Names ceasing to exist are not interesting. If a name is used 5 times in 1980, and never in 2013, it might have changed an infinite percentage, but doesn't get at the heart of this question.

With this in mind, below is a table of popularity changes, with various minimum popularity cutoffs. Where ties occured (such as with no minimum number of observations) the 'winner' is chosen arbitrarily:

| min_num_observations | popularity | biggest_change | num_occurrences_1980 | num_occurrences_2015 | perc_change | |----------------------|----------|----------------|----------------------|----------------------|----------------------| | 0 | increase | Ailani | 0.0 | 163.0 | inf | | 10 | increase | Isabella | 23.0 | 15504.0 | 673.086956522 | | 100 | increase | Sebastian | 121.0 | 9569.0 | 78.0826446281 | | 1000 | increase | Olivia | 1090.0 | 19553.0 | 16.9385321101 | | 10000 | increase | Benjamin | 13665.0 | 13608.0 | -0.00417124039517 | | 0 | decrease | Kenyetta | 70.0 | 0.0 | -1.0 | | 10 | decrease | Misty | 5541.0 | 14.0 | -0.997473380256 | | 100 | decrease | Heather | 19989.0 | 230.0 | -0.988493671519 | | 1000 | decrease | Jennifer | 58529.0 | 1247.0 | -0.978694322473 | | 10000 | decrease | Michael | 69173.0 | 14331.0 | -0.792823789629 |

Descriptive Analysis Q5

Q5) Can you identify names that may have had an even larger increase or decrease in popularity?

A5) I feel that this has been rolled in to q4. It's hard to get more than an infinite increase in popularity, or to more than completely go into extinction, so I'd say it is not possible to identify names that may have had an even larger increase or decrease in popularity.

Section B: Onward to Insight!

Q) Given the time commitment of the preceding sections, I've opted to go light on this section, and examine a few trends in baby naming that I'm curious about. Below are a few questions I've looked at.

Name variation

I was interested in seeing if it was possible to normalize name variations with off the shelf components. Ideally, this word normalize names, something like

- (Frank, Franky, Frankie, Franklin) -> Frank

I used NLTK's Porter Stemmer and WordNet Lemmatizer to attempt to normalize words. Unfortunately, spot checking showed that this normalization was not sufficient. It's likely possible to find / build a more robust method, but I've elected

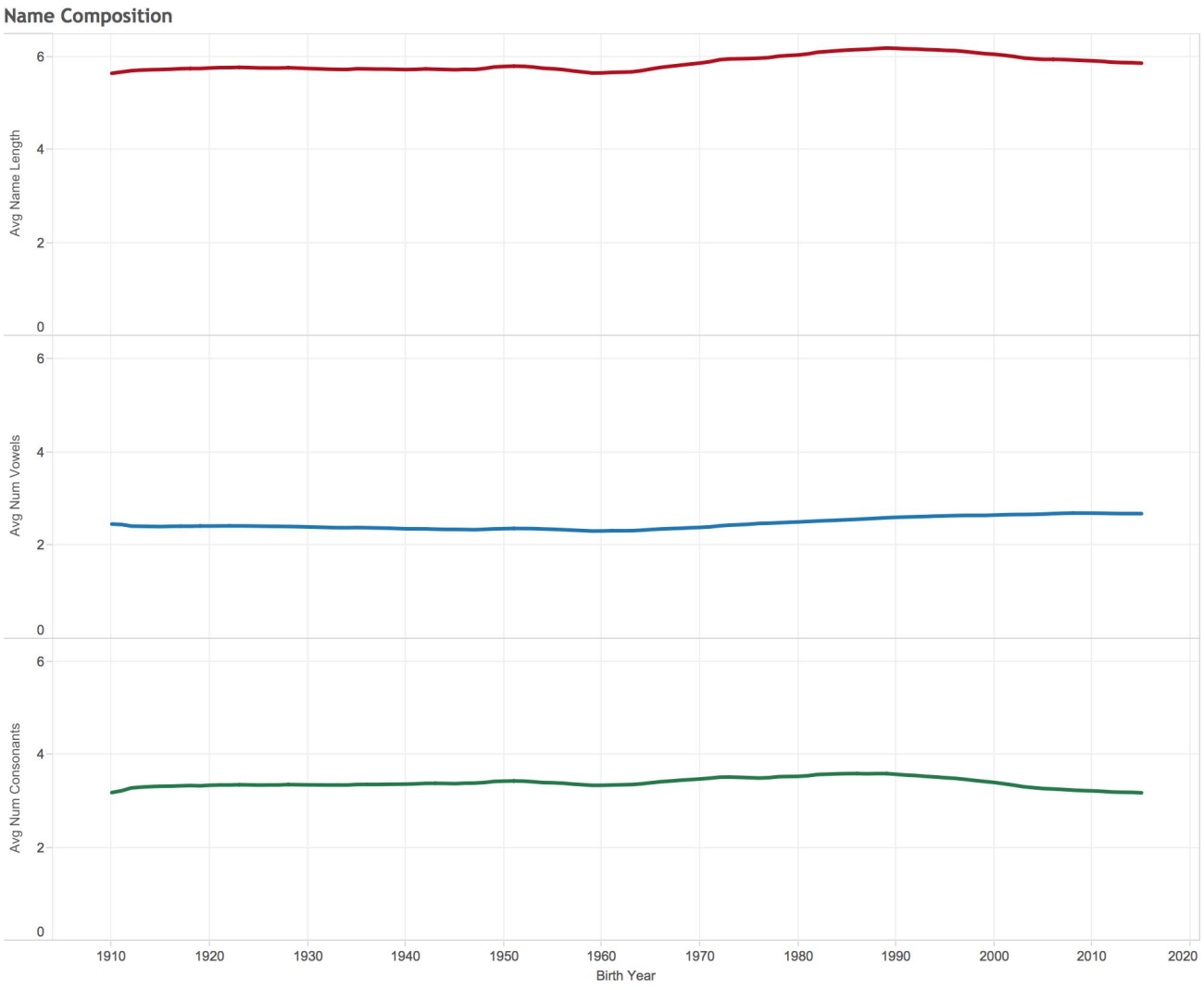
Composition Statistics

I was also interested in looking at how names sound / are composed, and how this changes year over year. A few details that I looked at were:

- Name length
- Number of vowels

- Number on consonants
- Percentage of vowels
- Whether names are English language words: This is difficult, because most dictionaries include common given names.

Below is a look at how name length, number of consonants and number of vowels have fared over time:



The trends of sum of Avg Name Length, sum of Avg Num Vowels and sum of Avg Num Consonants for Birth Year.

Interestingly, it looks like on average names have gotten slightly shorter and more vowel dense.