Predicting recipe sentiment from nutrient composition

Mike Touse

March 4, 2017

The goal of this project is to develop a model to predict the popularity of a given recipe to allow both restaurants and recipe publishers to better select content for their recipe collections before launching into costly customer trials and complicating their menu and associated supply chain requirements. In order to maximize extensibility into various markets, the model is developed using nutrient content as a basis (as opposed to ingredients, ethnicity, or other high-level features). After supervised training of the model with only nutrient content and customer ratings, new recipes were classified as either favorable or unfavorable with 75% precision.

Introduction

Upserve and Avero are among a number of restaurant management resources that help restaurants manage everything from employee schedules, customer history, and sales analytics. According to Upserve's website

> Every day, 14 million Americans show up to work at a restaurant. Thats 10% of the U.S. workforce. Almost all restaurants have fewer than 50 employees, and most are independent, not part of a giant chain. The restaurateurs leading these businesses run on tight margins. They dont have a big capital budget. They arent software gurus. And most of all, they dont have much time to get a handle on everything guests, staff, menu, marketing and finance making it harder than it has to be to take their restaurant to the next level.

One particular feature of Upserve's platform, which serves more than 7,000 restaurants nationwide, is 'Menu Intelligence' which couples sales data with menu items at the individual-customer level. Because sales trends are a lagging indicator, such management platforms (and their customers) could benefit tremendously from increased pre-launch confidence in recipe success.

Similarly, recipe websites generate revenue through either subscription fees or advertisement sales and rely heavily on customer satisfaction with the recipes presented. With

nearly 100 million monthly site visits to a top recipe website, improving site membership and click-through-rate (CTR) can generate significant revenue for minimal investment. Predicting recipe performance even before receiving user ratings would allow the company to highlight new recipes with reasonable confidence that the recipe would improve overall site utilization and yield higher revenue-generating CTRs.

Data Collection

In order to develop a precise and accurate model of recipe success, a collection of recipe nutrients and a customer rating of those recipes was required. Yummly.com uses two different Application Programming Interfaces (API's) to provide responses in JSON format:

- Search Recipe - returns a list of recipes that meet the search criteria and provide some attributes of the recipe
- Get Recipe - uses a single ID provided from the Search Recipe response to return the detailed nutritional data and recipe text details

Because each recipe must be downloaded individually once the list of recipes to download was selected, scripts were developed to handle the API requests within the education license term limits through the following process:

- Query subsets of recipes using individual ingredients (such as beef, chicken, tomatoes, etc). Responses included ratings and recipe id's.
- Concatenate subsets into single list of recipes.
- Query individual recipe id's from concatenated list to capture complete recipe information.
- Parse necessary fields into individual components (namely, the quantity of individual nutrients).

Following an initial collection of the recipe list, it became apparent that the recipe ratings were highly unbalanced (with nearly all recipes rated 4/5 and a smaller percentage rated 3/5 or 5/5 as seen in Figure 1). Further investigation suggested that selecting recipes towards the end of the API response tended towards lower ratings and the bulk collection scripts were adapted to balance the rating distribution of the dataset to allow improved model training.

Even after adding lower-ranked data to the set, most of the recipes were rated either 3/5 or 4/5. This would likely give poor results if attempting to predict actual recipe ratings, so the goal was shifted to a binary classification of either 'Favorable' (4/5 or 5/5) or 'Unfavorable' (0/5 - 3/5) whose final distribution is shown in Figure 2.

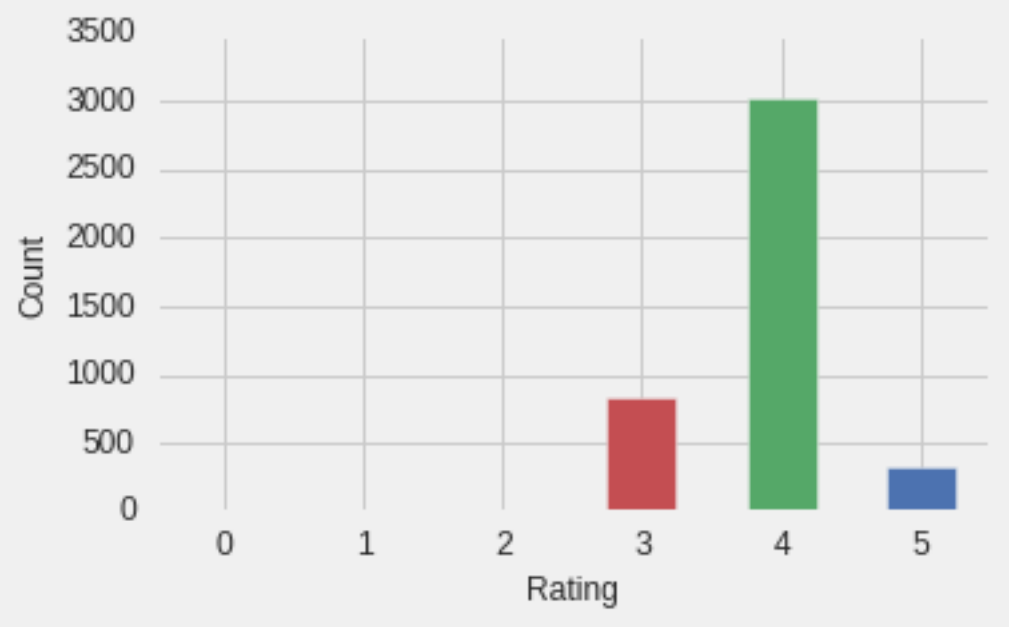

Figure 1: Distribution of recipe ratings following initial download.

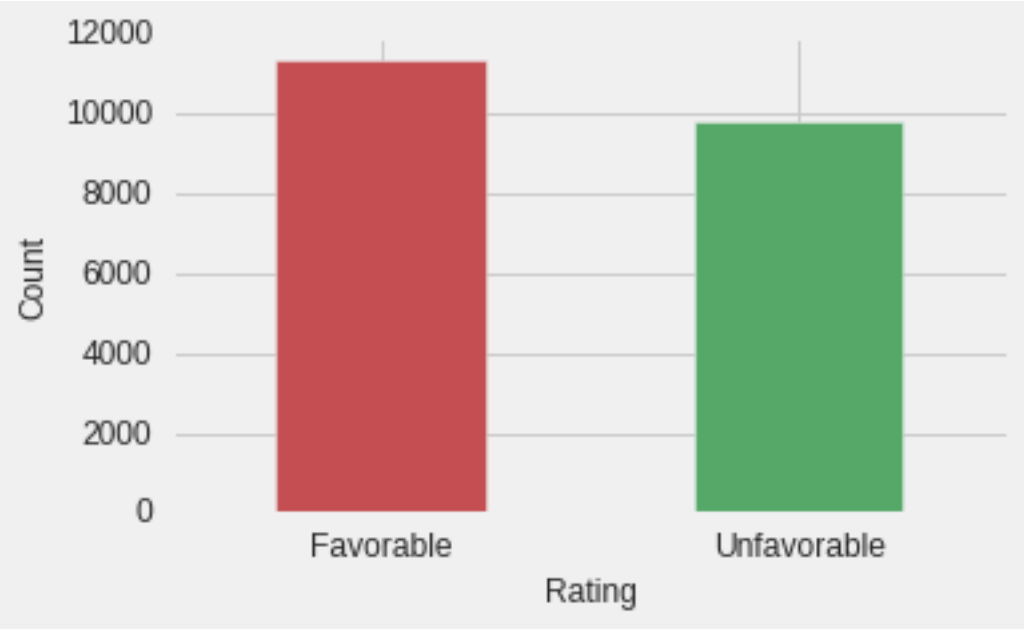

Figure 2: Distribution of recipe ratings used for model development. Note roughly equal shares of Favorable vs Unfavorable ratings.

Table 1: Descriptive statistics of a sampling of features from final dataset. It is important to note that many features remain 0.0 through multiple quartiles, indicating sparsity of many nutrients.

	FAT_KCAL	K	FLD	PHYSTR	VAL_G
mean	51.6	0.16	0.0000	0.005	0.19
std	94.8	0.23	0.0000	0.015	0.34
min	0.0	0.00	0.0000	0.000	0.00
25%	4.2	0.01	0.0000	0.000	0.00
50%	23.3	0.08	0.0000	0.001	0.04
75%	62.5	0.21	0.0000	0.005	0.28
max	1810.0	2.95	0.0025	0.630	5.54

Data Structure

Once the recipe list was compiled using the Search Recipe API, the individual recipes were downloaded and the JSON responses were ingested into a Pandas DataFrame. The nutrient information was contained in a list within a single column and required further parsing. Ultimately, a DataFrame was compiled that contained the recipe ID, rating, number of servings, preparation time, and individual nutrients, each as a separate feature (column). The recipe response only populates fields for the nutrients listed for the individual recipe, so the DataFrame was constructed to include every nutrient listed, with many of the fields populated with null ('NaN') values. Nutrient values were normalized to a 'per serving' value using the 'number of servings' field, and null elements were filled with a value of '0'.

Data Exploration

The final data set after cleaning included 21,140 samples (recipes) comprised of 113 nutrients (with many nutrients not present in all recipes). A sampling of five nutrients are presented in Table 1 along with some descriptive statistics such as mean, standard deviation, and quartile values. The list of all possible nutrients and their abbreviations are presented in the Appendix (Table 4).

Initial exploratory data analysis focused on examining which of the nutrients were reported in the largest percentage of recipes and the distribution of ratings across each feature in order to identify potential predictive features. Results from this analysis revealed very little correlation between any particular nutrient and the rating/class label, as illustrated in Figure 3 which shows the correlation coefficient across all features, with the vast majority of nutrients negatively correlated with the recipe classification. The maximum (absolute) correlation coefficient was found to be 0.18 for K (Potassium, which is negatively correlated).

The lack of correlation between nutrients can be seen even more clearly in Figure 4. The nutrients shown were determined to be some of the most important features in

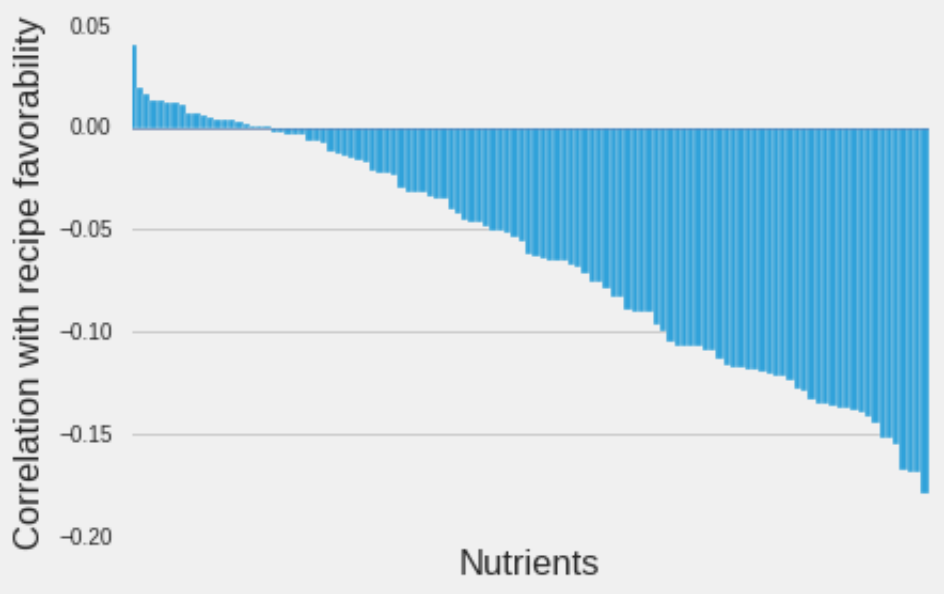

Figure 3: Correlation coefficient of all nutrients with positive recipe classification. Note low levels of correlation and negative skew.

the classification methods used in following sections and demonstrate the lack of any separating plane between the positively rated (red) and negatively rated (blue) recipes, likely preventing the use of linear discriminants or support vectors for classification. The visual presentation in Figure 4 does, however, support the observation in Figure 3 that most nutrient quantities are negatively correlated with recipe classification.

Each nutrient was then examined to determine whether a particular value of that nutrient (per serving) would yield a significant favorability descriminant. By viewing the fraction of positive ratings in each decile, it was noted that most nutrients followed a downward trend with increasing nutrient content but with a significant minimum within a particular decile. Nutrients were then grouped by the location of their minimum favorability decile and plotted as the portion of positive ratings in each decile versus the actual decile. Line widths demonstrate the number of nutrients in each grouping which indicates that many recipes become unfavorable as the nutrient quantities increase above roughly their fifth or sixth decile, though many others demonstrate minima at lower deciles.

Classification

Logistic Regression

In order to identify weights of particular features that can be used to classify individual recipes, the first step was to attempt to fit a logistic regression to the nutrient data.

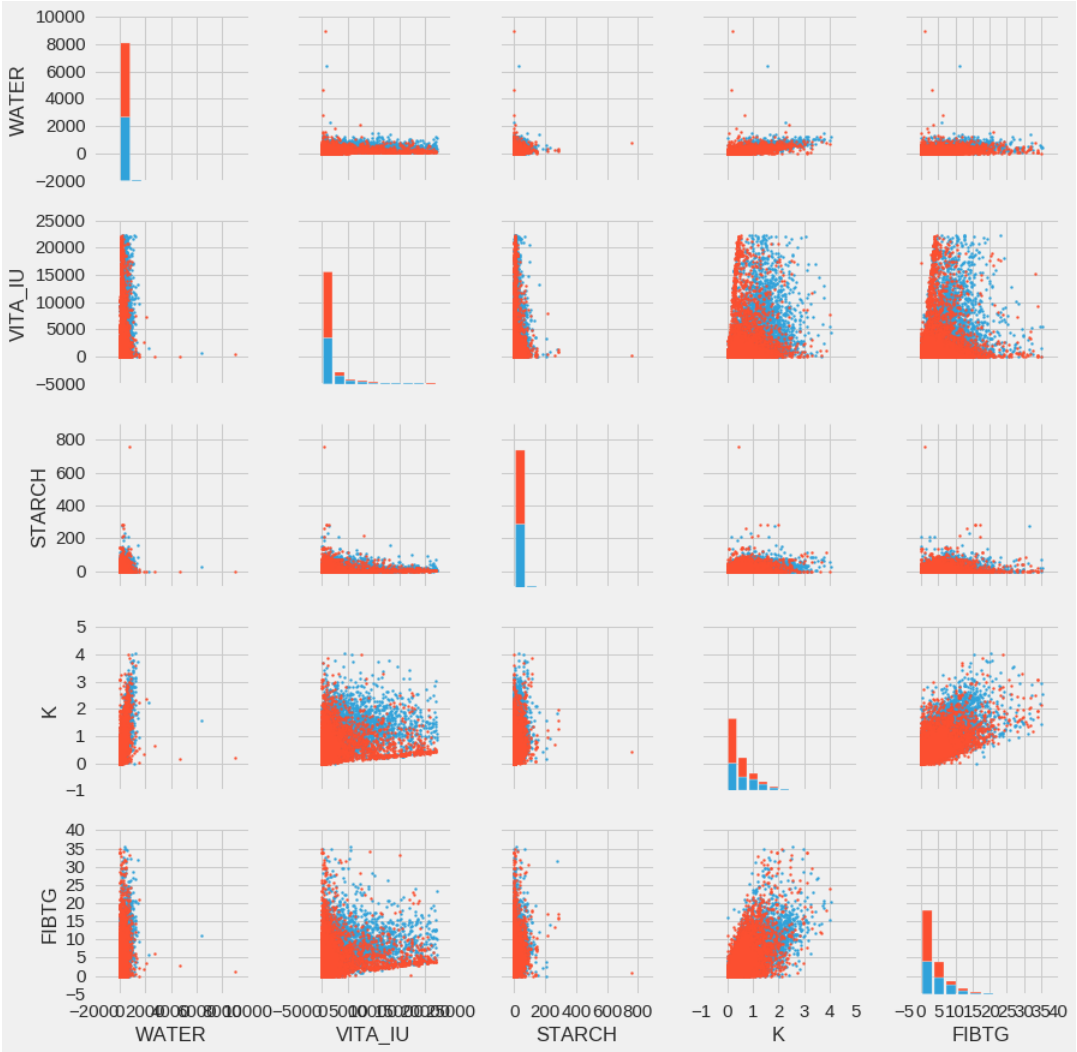

Figure 4: Scatter plots of several nutrient pairs (red points are highly rated recipes, blue are poorly rated). Plots shown are illustrative of the data distribution across the entire dataset. Diagonal plots show histogram of the given feature.

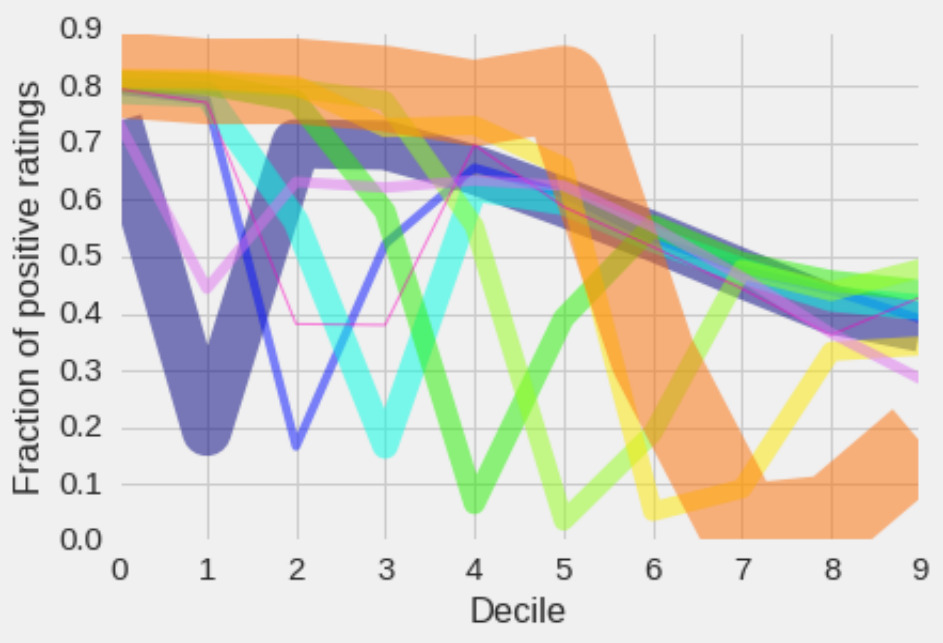

Figure 5: Fraction of recipes that are highly rated at each decile, by nutrient, grouped according to decile that contains the minimum favorability. Line widths indicate number of nutrients in each group.

Because of the large number of features and relative sparsity of some, L1 regularization was expected to yield more useful results by driving a portion of individual feature weights to zero.

To prepare the data to develop a logistic regression classifier, the data was scaled using the scikit-learn StandardScaler, and data was split into randomly sampled training (80%) and test (20%) recipes. Initial classification with Logistic Regression was attempted with C=1 and L1 regularization with successful convergence and acceptable results. Several other values of C were tested with little impact to accuracy and other metrics. L2 regularization was also tested with little impact.

The logistic regression classifier was able to predict recipe classification with approximately 66% accuracy (F1 = 73%). These and other metrics are shown in Table 2. As can

Table 2: Performance metrics of logistic regression classification.

Metric	**Value**
Accuracy	0.66
Precision	0.64
Recall	0.87
F1	0.73
Area under ROC curve	0.70

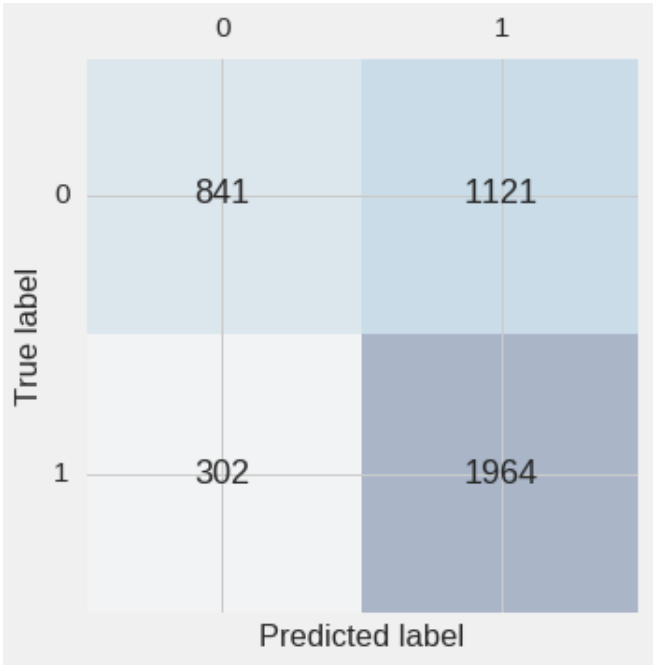

Figure 6: Confusion matrix of predicted classification of test samples following logistic regression classifier supervised training. Labels of '0' indicate low ratings and labels of '1' indicate high ratings.

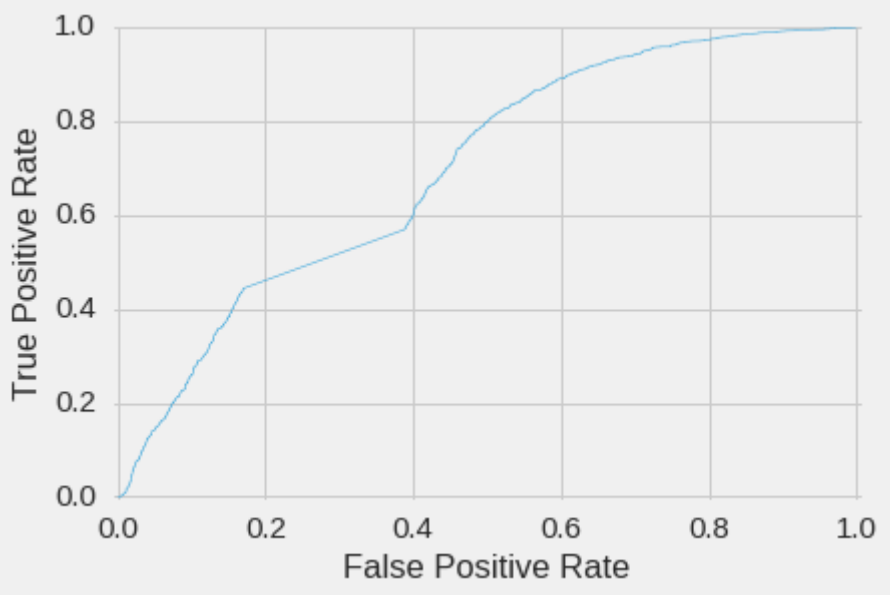

Figure 7: Receiver Operating Characteristic curve showing True Positve Rate versus False Positive Rate following logistic regression classification.

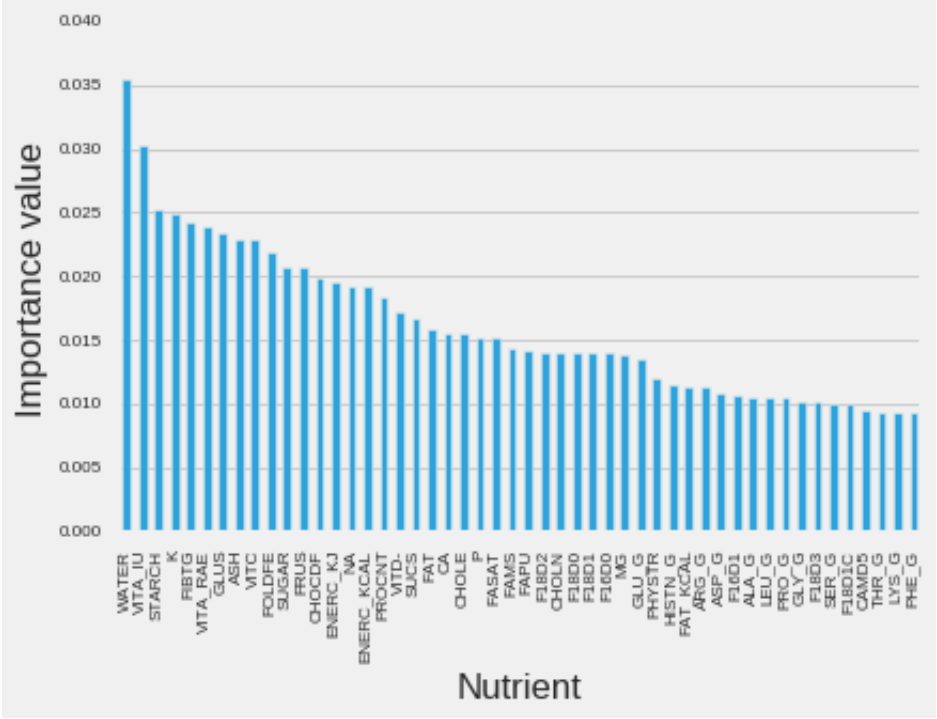

Figure 8: Fifty most important features identified in Random Forest classifier.

be seen in the confusion matrix (Figure 6), the logistic regression classifier suffered from a significant number of false positive classifications. While the developed model would offer slight benefit to the example restaurateur or recipe author, better performance was expected through ensemble methods such as a Random Forest classifier which is reported in the next section.

Random Forest Classifier

Because the Random Forest classifier samples across all features and recipes to build each individual tree, the original training dataset (scaled only to single-recipe values) was used instead of the set that was fit using the StandardScaler for logistic regression. Using the scikit-learn Random Forest Classifier with 10,000 estimators and a maximum of 11 features per tree, the classifier was trained and feature importance was determined, the top 50 of which are shown in Figure 8.

After fitting the training data, the test data was again used to predict the ratings and compared to the true ratings from the original dataset. Similar to the logistic regression classifier reported above, the performance of the Random Forest classifier was evaulated using the metrics reported in Table 3. Significant improvement was seen as compared to the performance of the logistic regression, with accuracy of 74%. All metrics improved with the exception of recall, which benefitted from the high False Positive Rate in the logistic regression classifier by capturing a large percentage (87%) of the positives with

Table 3: Performance metrics of Random Forest classification.

Metric	Value
Accuracy	0.74
Precision	0.75
Recall	0.78
F1	0.76
Area under ROC curve	0.80

its high positive bias.

Examination of the confusion matrix (Figure 9) shows that the correctly predicted recipes are much more evenly distributed between true positive (1, or high ratings) and true negative (0, or low ratings), and correct predictions far outnumber incorrect as seen in the metrics above. Additionally, the ROC curve in Figure 10 shows a better balance of True and False Positive Rates, yielding an area of 0.80 under the ROC curve.

Because the random forest classifier aggregates a large set of decision trees, it is important to illustrate that the relative importance values do not equate to relative concentrations, nor do they identify the relative importances of any particular decision tree. In Figure 11, the importance values of the individual nutrients in a single recipe are presented with the top 50 aggregate importances shown previously in Figure 8. Note that the (orange) individual tree values show a similar trend across the aggregate values, but individual nutrient importances differ greatly when considered in a single recipe.

Finally, it is worth discussing overall observations of the nutrients identified as important for classifying recipe likability. As discussed earlier, the most strongly identified nutrients have negative correlations, meaning that excess amounts of those nutrients are most likely to cause a recipe to be rated poorly.

Some very familiar nutrients stand out as being significant and negatively correlated, namely water, potassium, vitamin C, and vitamin A. Another observation is a rough grouping of the amino acids such as alanine, serine, histidine, and glutamic acid near the top third of the identified nutrients (again, negatively correlated). While no causation stands out that explains the observations, one hypothesis is that some of these nutrients might cause a recipe to be assessed as too strong (vitamins A and C) or too diluted (water) causing harsher critique by the rater.

Conclusion

In the analysis presented, a set of rated recipes and their nutrients was collected, cleaned, and evaluated for potential discriminating features to allow prediction of user rating. The data was then used to ultimately train a Random Forest classifier which correctly predicted user rating with 74% accuracy.

Preselecting recipes, either as a restaurateur or recipe author, based on a reasonable certainty of their success is obviously advantageous in terms of time and resource efficiency. An additional advantage of the analysis and supporting data is that individual

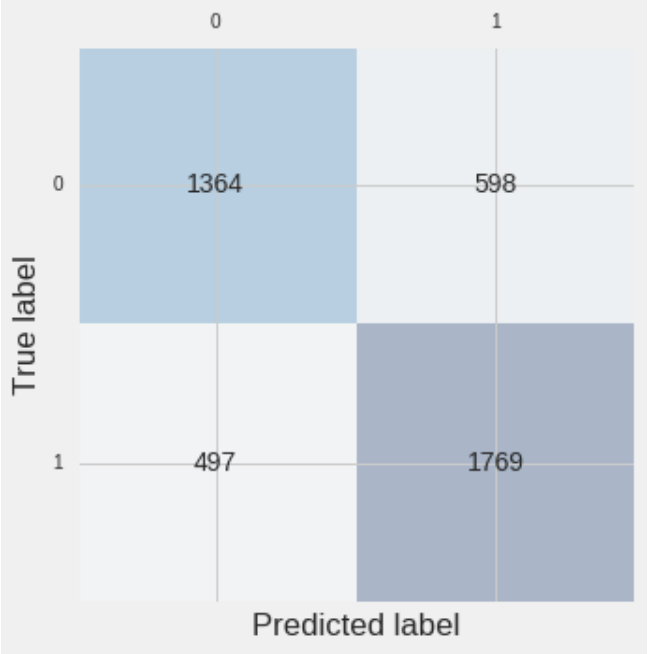

Figure 9: Confusion matrix of predicted classification of test samples following Random Forest classifier supervised training. Labels of '0' indicate low ratings and labels of '1' indicate high ratings.

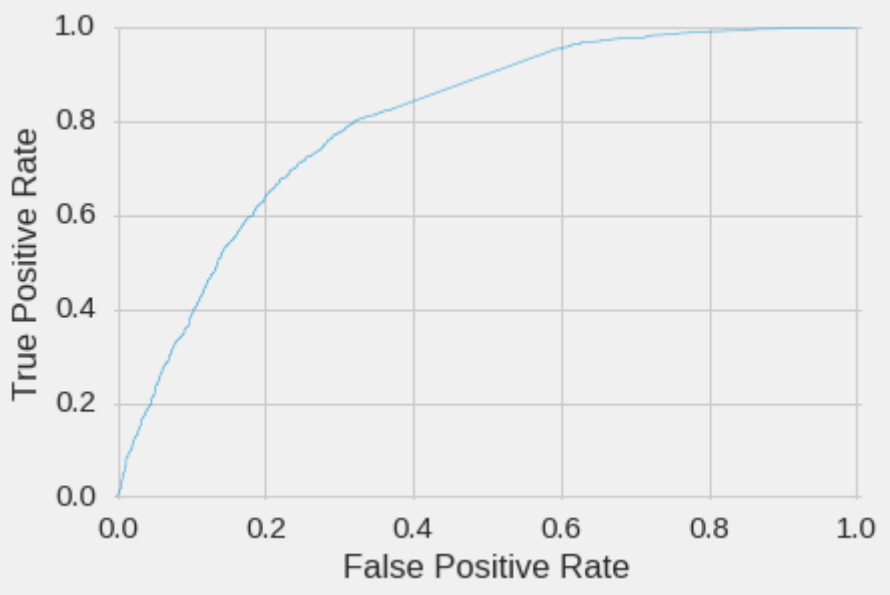

Figure 10: Receiver Operating Characteristic (ROC) curve of Random Forest classifier output. Area Under ROC Curve (AUC) = 0.80.

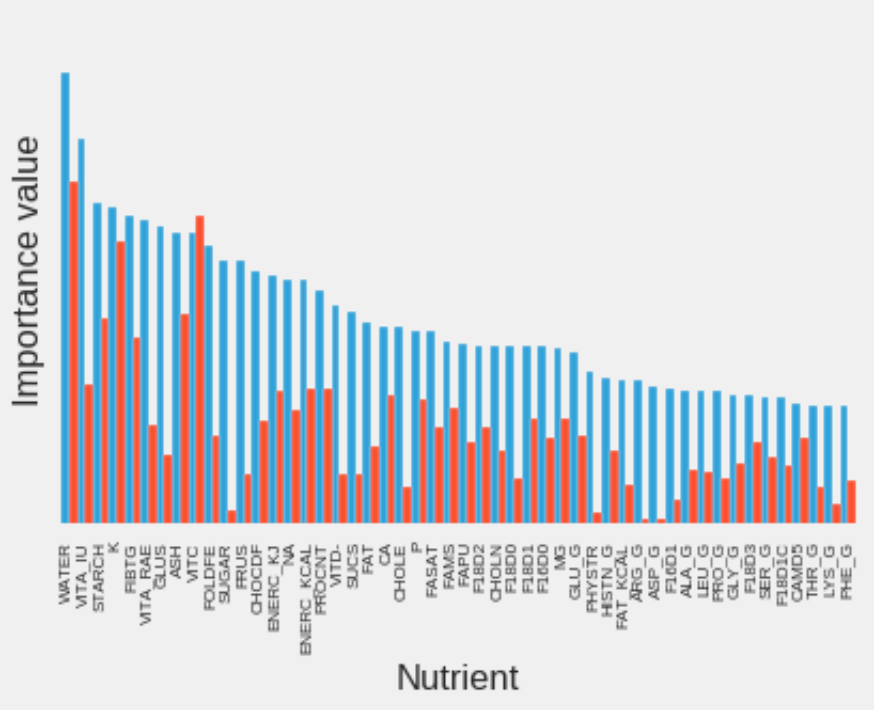

Figure 11: Feature importances for top 50 features from Random Forest classifier (blue), with importances from a single test tree with high probability of positive rating (orange). Actual importance values are relative and, therefore, not shown.

nutrients were identified that can provide the user guidelines within which they can develop recipes that have, at a minimum, 74% chance of being rated favorably by users or customers.

The closing advice to a recipe developer would be to strongly consider the most positively correlated nutrients: caffeine and theobromine which can both be found in high concentrations in chocolate.

Appendix: Nutrient List

Nutrient	Units	Tagname
Protein	g	PROCNT
Total lipid (fat)	g	FAT
Carbohydrate, by difference	g	CHOCDF
Ash	g	ASH
Energy	kcal	ENERC_KCAL
Starch	g	STARCH
Sucrose	g	SUCS
Glucose (dextrose)	g	GLUS
Fructose	g	FRUS
Lactose	g	LACS
Maltose	g	MALS
Alcohol, ethyl	g	ALC
Water	g	WATER
Adjusted Protein	g	NaN
Caffeine	mg	CAFFN
Theobromine	mg	THEBRN
Energy	kJ	ENERC_KJ
Sugars, total	g	SUGAR
Galactose	g	GALS
Fiber, total dietary	g	FIBTG
Calcium, Ca	mg	CA
Iron, Fe	mg	FE
Magnesium, Mg	mg	MG
Phosphorus, P	mg	P
Potassium, K	mg	K
Sodium, Na	mg	NaN
Zinc, Zn	mg	ZN
Copper, Cu	mg	CU
Fluoride, F	g	FLD
Manganese, Mn	mg	MN
Selenium, Se	g	SE
Vitamin A, IU	IU	VITA_IU
Retinol	g	RETOL
Vitamin A, RAE	g	VITA_RAE
Carotene, beta	g	CARTB
Carotene, alpha	g	CARTA
Vitamin E (alpha-tocopherol)	mg	TOCPHA
Vitamin D	IU	VITD
Vitamin D2 (ergocalciferol)	g	ERGCAL

Vitamin D3 (cholecalciferol)	g	CHOCAL
Vitamin D (D2 + D3)	g	VITD
Cryptoxanthin, beta	g	CRYPX
Lycopene	g	LYCPN
Lutein + zeaxanthin	g	LUT+ZEA
Tocopherol, beta	mg	TOCPHB
Tocopherol, gamma	mg	TOCPHG
Tocopherol, delta	mg	TOCPHD
Tocotrienol, alpha	mg	TOCTRA
Tocotrienol, beta	mg	TOCTRB
Tocotrienol, gamma	mg	TOCTRG
Tocotrienol, delta	mg	TOCTRD
Vitamin C, total ascorbic acid	mg	VITC
Thiamin	mg	THIA
Riboflavin	mg	RIBF
Niacin	mg	NIA
Pantothenic acid	mg	PANTAC
Vitamin B-6	mg	VITB6A
Folate, total	g	FOL
Vitamin B-12	g	VITB12
Choline, total	mg	CHOLN
Menaquinone-4	g	MK4
Dihydrophylloquinone	g	VITK1D
Vitamin K (phylloquinone)	g	VITK1
Folic acid	g	FOLAC
Folate, food	g	FOLFD
Folate, DFE	g	FOLDFE
Betaine	mg	BETN
Tryptophan	g	TRP_G
Threonine	g	THR_G
Isoleucine	g	ILE_G
Leucine	g	LEU_G
Lysine	g	LYS_G
Methionine	g	MET_G
Cystine	g	CYS_G
Phenylalanine	g	PHE_G
Tyrosine	g	TYR_G
Valine	g	VAL_G
Arginine	g	ARG_G
Histidine	g	HISTN_G
Alanine	g	ALA_G
Aspartic acid	g	ASP_G
Glutamic acid	g	GLU_G
Glycine	g	GLY_G

Proline	g	PRO_G
Serine	g	SER_G
Hydroxyproline	g	HYP
Vitamin E, added	mg	NaN
Vitamin B-12, added	g	NaN
Cholesterol	mg	CHOLE
Fatty acids, total trans	g	FATRN
Fatty acids, total saturated	g	FASAT
4:0	g	F4D0
6:0	g	F6D0
8:0	g	F8D0
10:0	g	F10D0
12:0	g	F12D0
14:0	g	F14D0
16:0	g	F16D0
18:0	g	F18D0
20:0	g	F20D0
18:1 undifferentiated	g	F18D1
18:2 undifferentiated	g	F18D2
18:3 undifferentiated	g	F18D3
20:4 undifferentiated	g	F20D4
22:6 n-3 (DHA)	g	F22D6
22:0	g	F22D0
14:1	g	F14D1
16:1 undifferentiated	g	F16D1
18:4	g	F18D4
20:1	g	F20D1
20:5 n-3 (EPA)	g	F20D5
22:1 undifferentiated	g	F22D1
22:5 n-3 (DPA)	g	F22D5
Phytosterols	mg	PHYSTR
Stigmasterol	mg	STID7
Campesterol	mg	CAMD5
Beta-sitosterol	mg	SITSTR
Fatty acids, total monounsaturated	g	FAMS
Fatty acids, total polyunsaturated	g	FAPU
15:0	g	F15D0
17:0	g	F17D0
24:0	g	F24D0
16:1 t	g	F16D1T
18:1 t	g	F18D1T
22:1 t	g	F22D1T
18:2 t not further defined	g	NaN
18:2 i	g	NaN

18:2 t,t	g	F18D2TT
18:2 CLAs	g	F18D2CLA
24:1 c	g	F24D1C
20:2 n-6 c,c	g	F20D2CN6
16:1 c	g	F16D1C
18:1 c	g	F18D1C
18:2 n-6 c,c	g	F18D2CN6
22:1 c	g	F22D1C
18:3 n-6 c,c,c	g	F18D3CN6
17:1	g	F17D1
20:3 undifferentiated	g	F20D3
Fatty acids, total trans-monoenoic	g	FATRNM
Fatty acids, total trans-polyenoic	g	FATRNP
13:0	g	F13D0
15:1	g	F15D1
18:3 n-3 c,c,c (ALA)	g	F18D3CN3
20:3 n-3	g	F20D3N3
20:3 n-6	g	F20D3N6
20:4 n-6	g	F20D4N6
18:3i	g	NaN
21:5	g	F21D5
22:4	g	F22D4
18:1-11 t (18:1t n-7)	g	F18D1TN7

Table 4: Table of all reportable nutrients as identified in the USDA Nutrition Database.