Renthop Kaggle

Daka for Data

Team

Domingos Lopes - Theoretical Mathematics and Academic

Abhishek Desai - Law Firm and Project Management

Arjun Singh Yadav - Robotics and Automation

Kamal Sandhu - Finance and Business

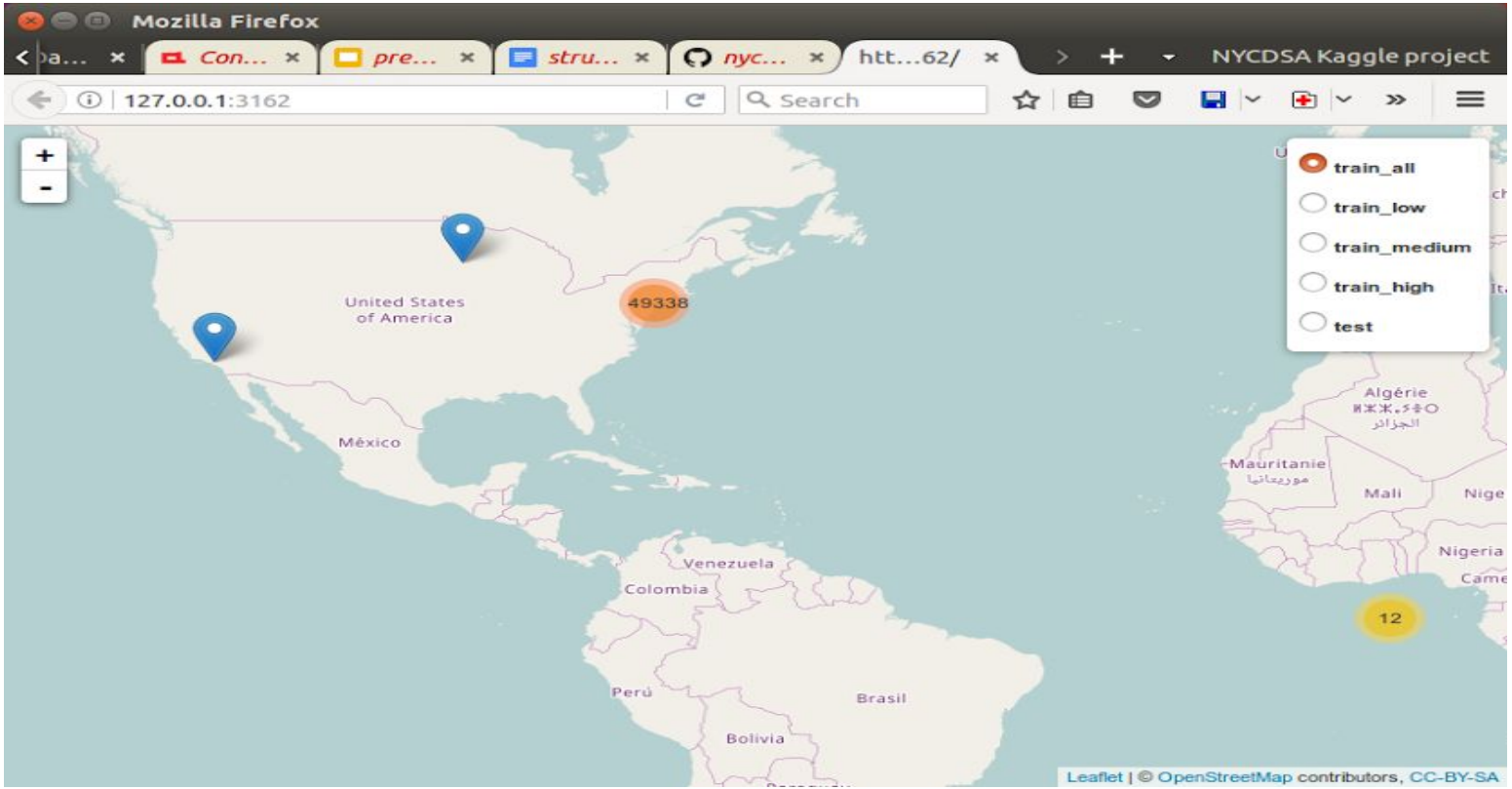
Team

- Combination of individual and team approach
- Everyone contributed to the project
- Brought different skills to the table
- Hard working and harmonious
- Focussed on learning rather than competing amongst us

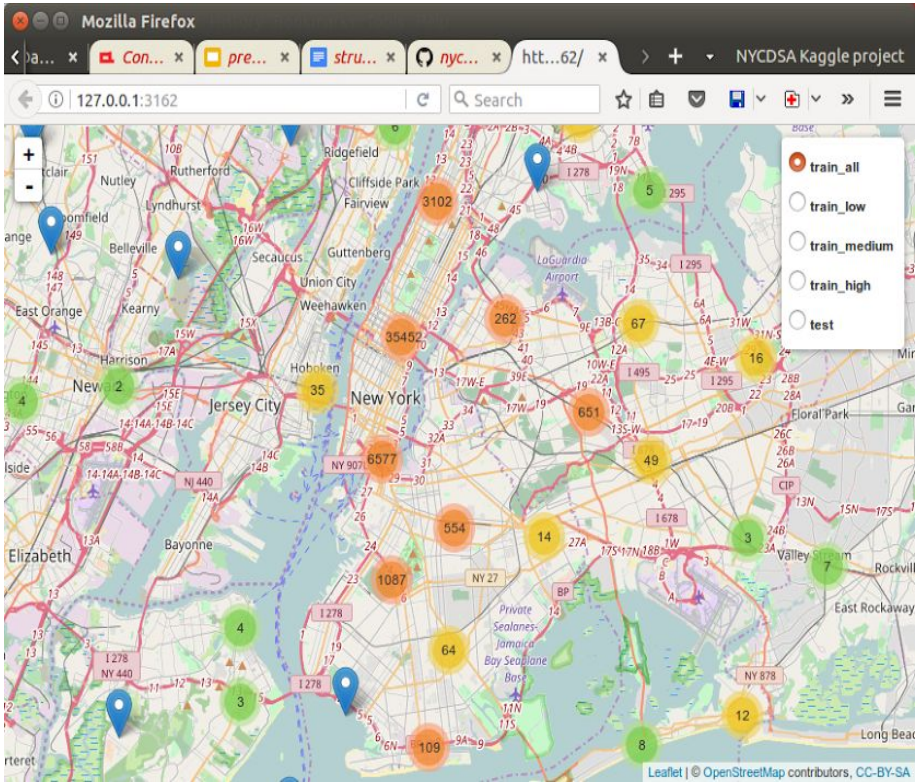
Two Sigma - Renthop

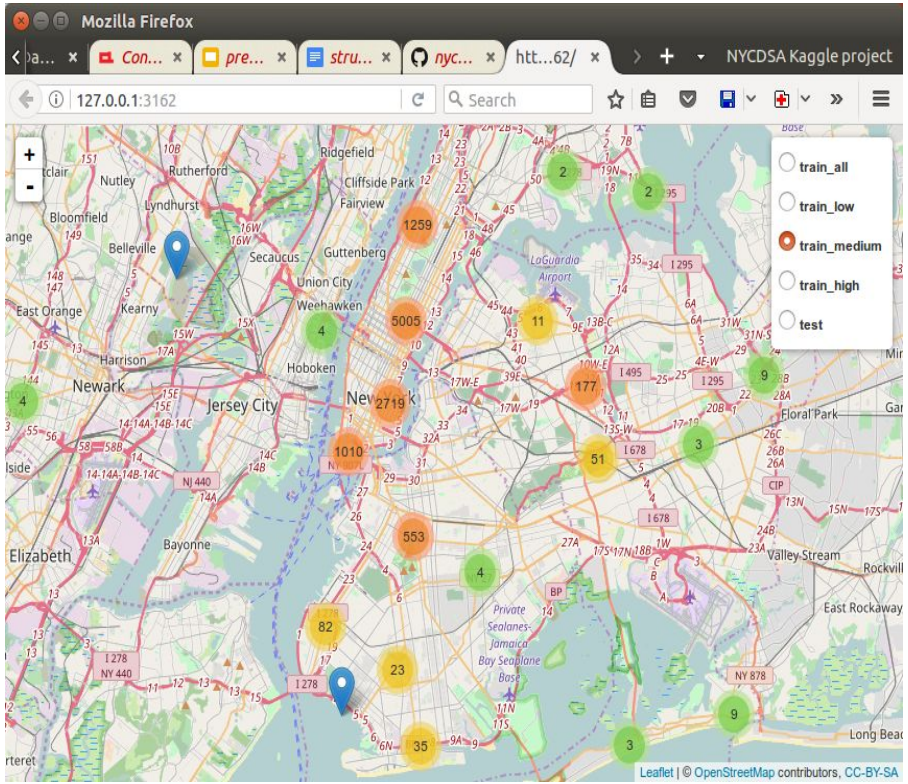
- Predict interest level in a rental listings based on feature fields derived from the listing posting on Renthop.com
- Approx. 49k labeled observations and 74k test observations
- Mix of structured/unstructured and categorical/numeric variables
- Logarithmic (cross-entropy) loss as the performance measurement function

Mapping App for EDA



Mapping App for EDA





Preliminary Feature Engineering

- Counted various details like number of pictures, number of features, etc
- For descriptions
  - Extracted buzzwords
  - Did sentiment analysis
  - n-gram extraction and scoring with sklearn text analysis framework
  - Feature agglomeration and Boltzmann machines to reduce dimensionality
- For features
  - Reduced all features to 55 categories
  - Count vectorized using custom code

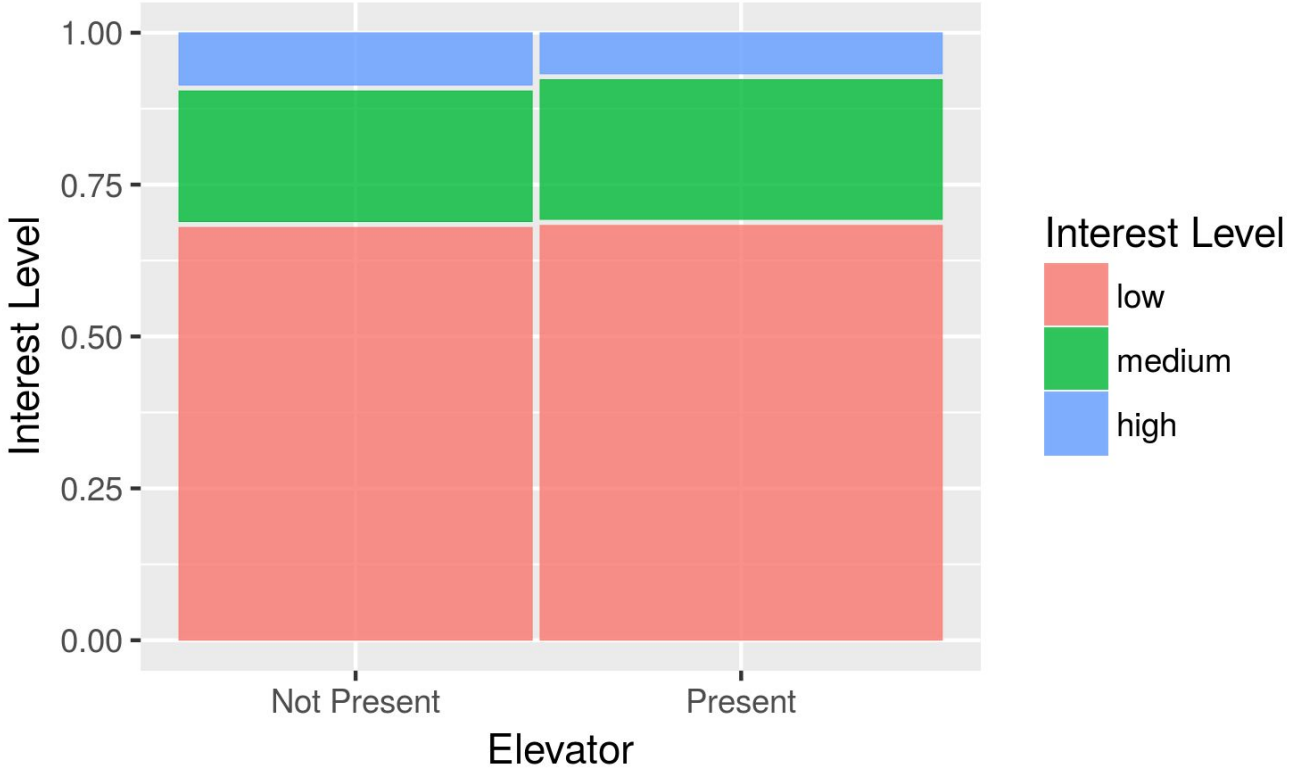
Preliminary Feature Engineering

- For lat/longs
  - Clustered them into groups of points
  - Clustering done based on their location within squares on a grid
- For categorical variables
  - One-hot encoding
  - Clustering to combine and make dataframes denser

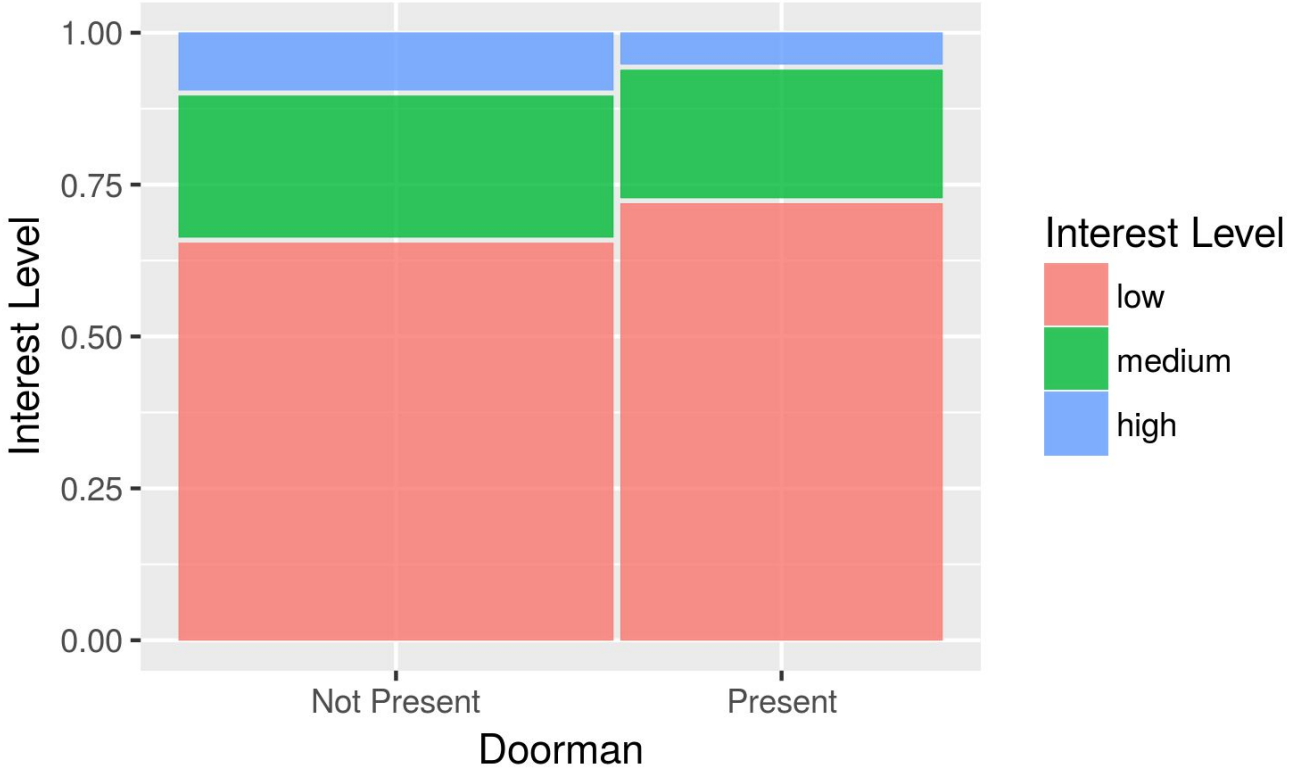
Packages and Libraries Used

- Scikit-learn - Extensive python package supporting various preprocessing, machine learning, ensemble and feature engineering classes
- NLTK - Text analytics
- TensorFlow - Numerical computations using graph architecture for neural networks
- Keras - High level neural network library using Theano and Tensoflow backend
- MLX Tend - Model ensembling
- XGBoost - Gradient boosting trees
- Caret - Ensembling and grid searching in R
- Syuzhet - Sentiment analysis

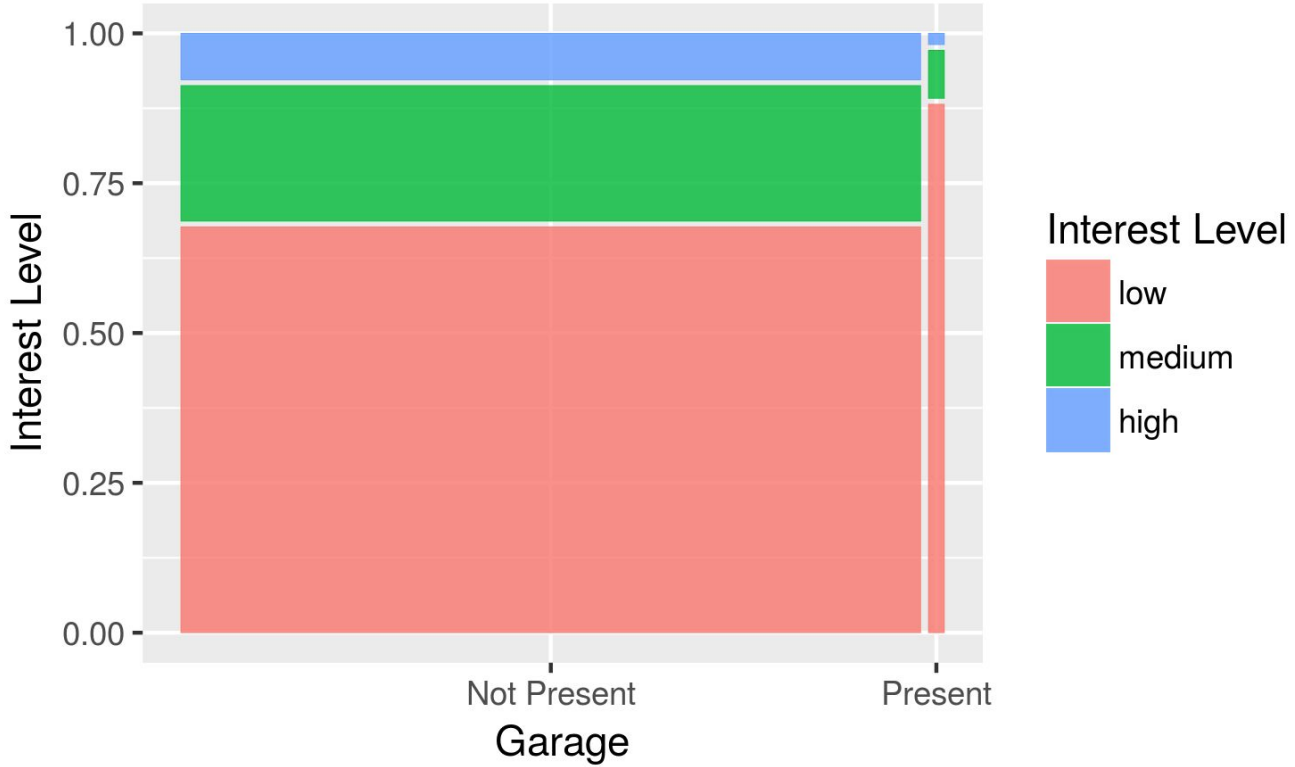
Elevator


1.00
0.75
0.50
0.25
0.00
Interest Level
Not Present
Present
Elevator
Interest Level
low
medium
high

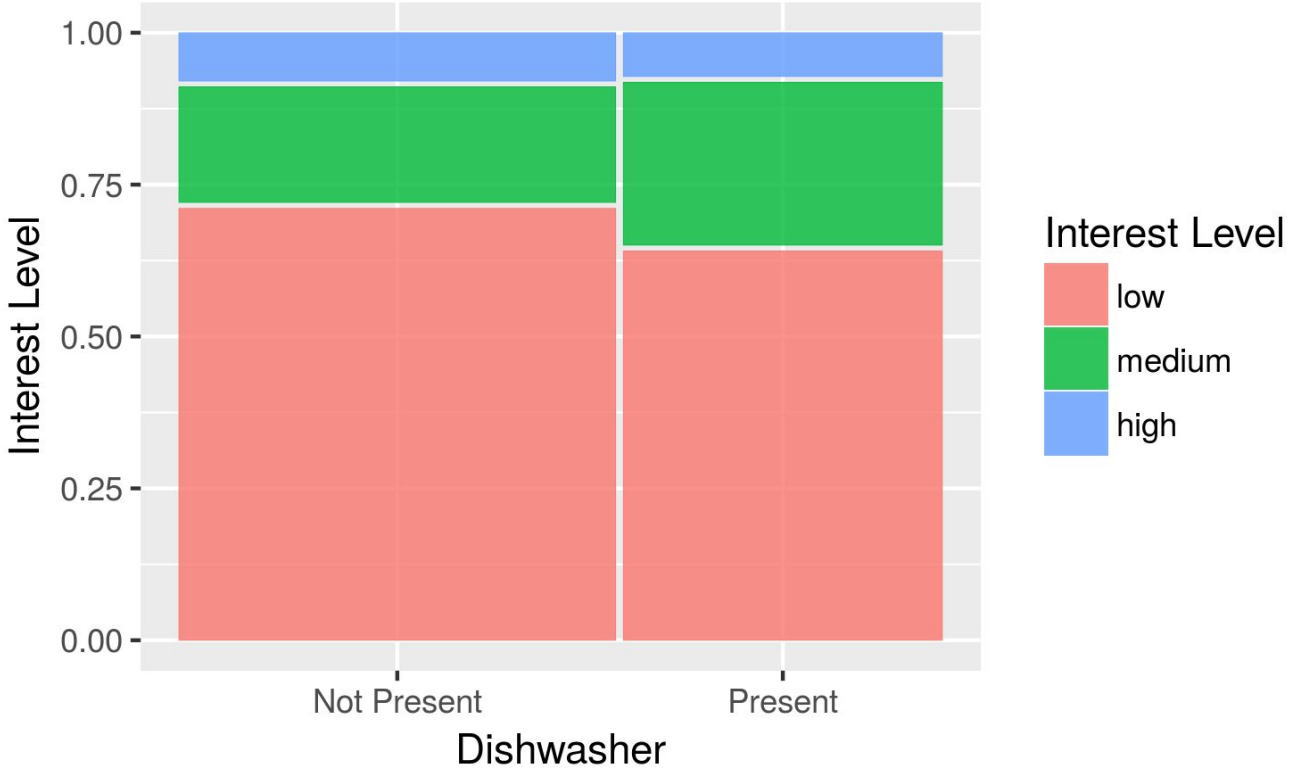
Doorman


1.00
0.75
0.50
0.25
0.00
Interest Level
Not Present
Present
Doorman
Interest Level
low
medium
high

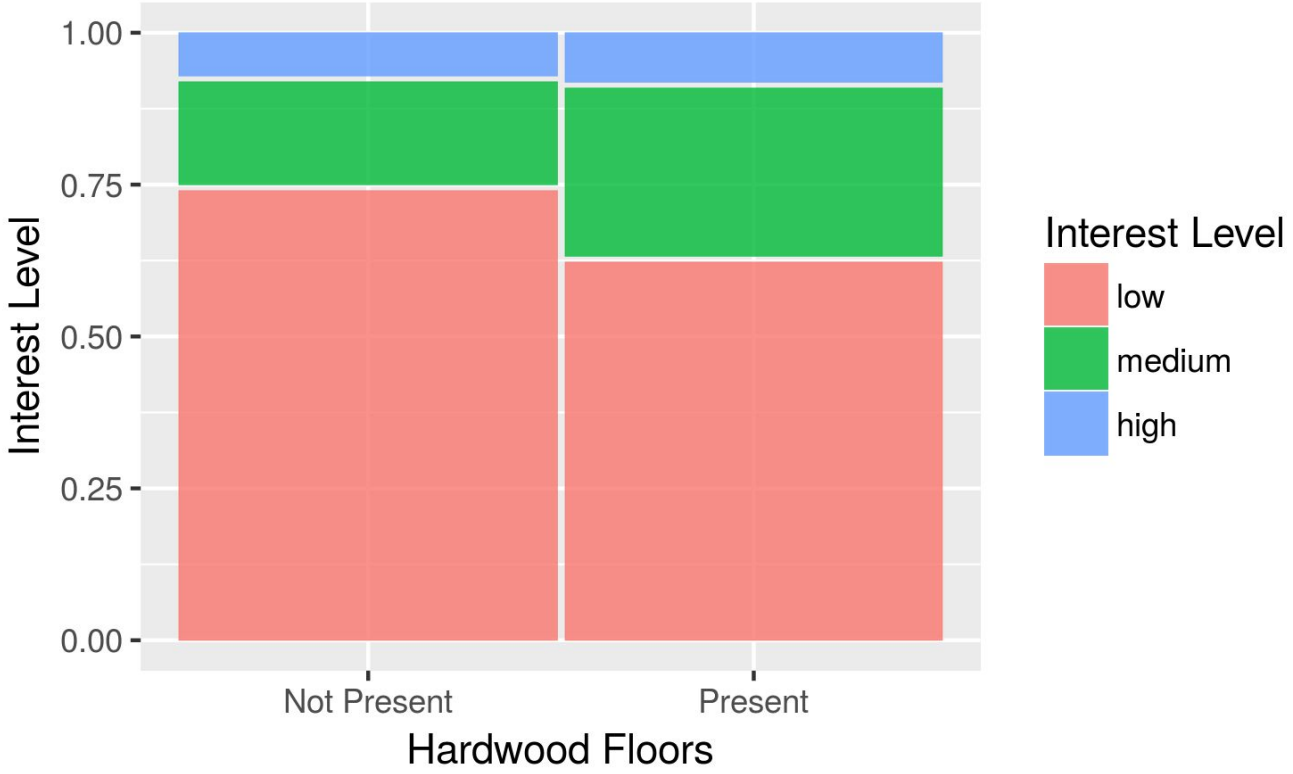
Garage


1.00
0.75
0.50
0.25
0.00
Interest Level
Not Present
Present
Garage
Interest Level
low
medium
high

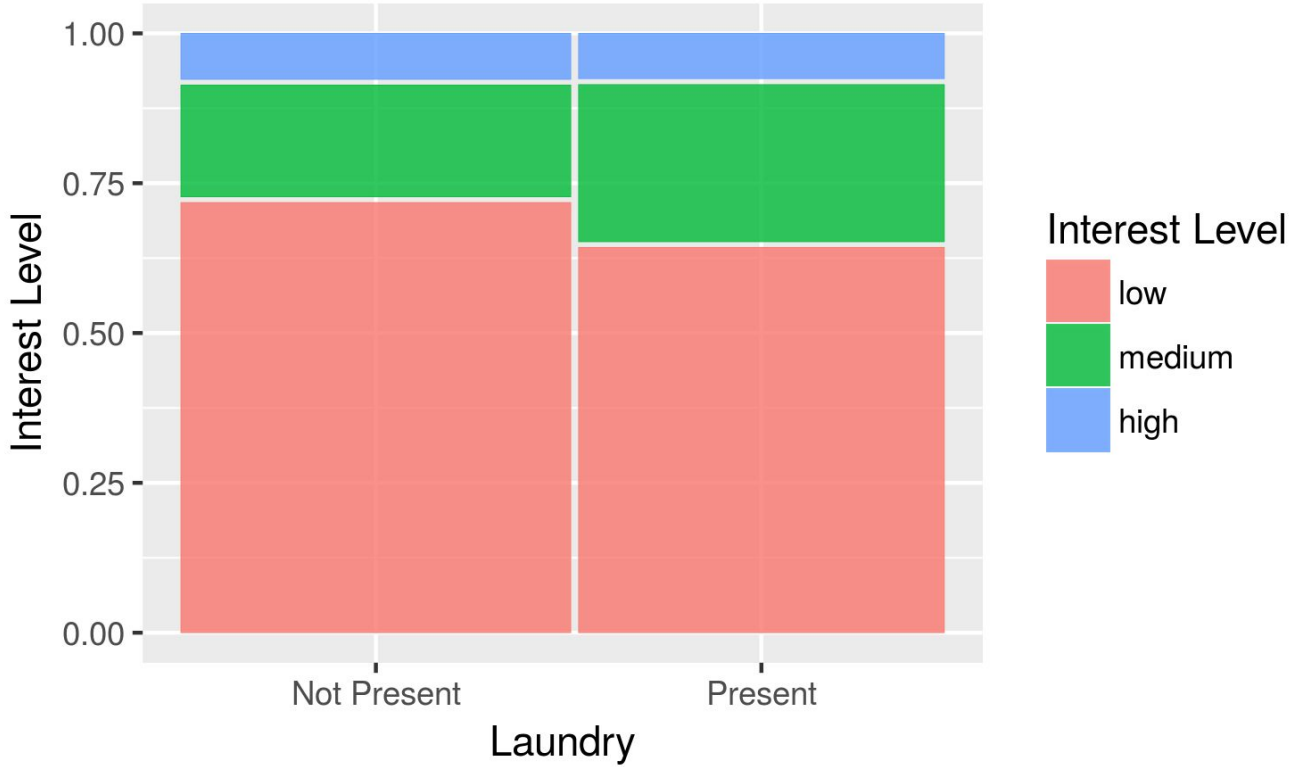
Dishwasher


1.00
0.75
0.50
0.25
0.00
Interest Level
Not Present
Present
Dishwasher
Interest Level
low
medium
high

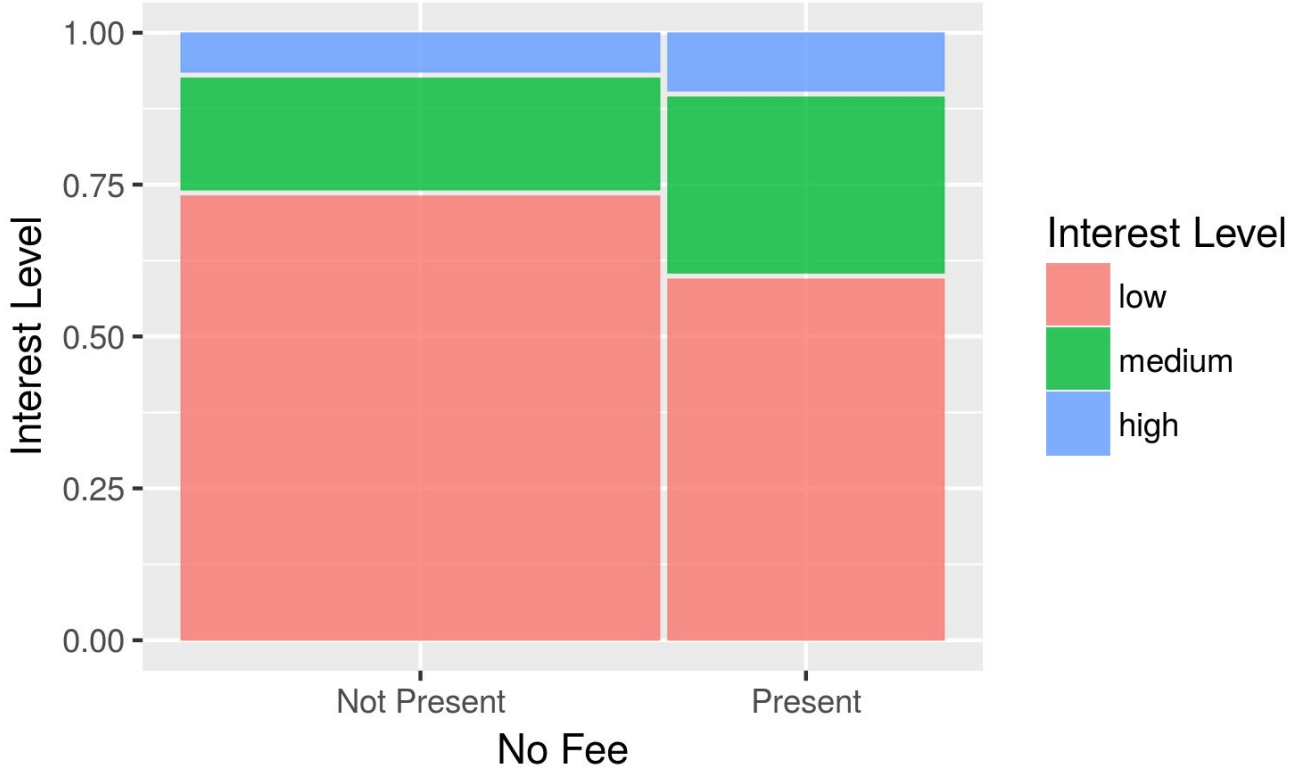
Hardwood Floors



Laundry


1.00
0.75
0.50
0.25
0.00
Interest Level
Not Present
Present
Laundry
Interest Level
low
medium
high

No Fee


1.00
0.75
0.50
0.25
0.00
Interest Level
Not Present
Present
No Fee
Interest Level
low
medium
high

Renovated


1.00
0.75
0.50
0.25
0.00
Interest Level
Not Present
Present
Renovated
Interest Level
low
medium
high

Question:

Will photographs in Renthop.com determine a increase or decrease in interest for a new listing posted?

Problems we might face:

- Size of the file
- Corrupt photos (90), Similar Photos (4752), No photos (265).

3466	6814336.0	6814336_df5b9f6282769a625025954a7eadff8c.jpg	640.0	373.0
32230	7248188.0	6814336_df5b9f6282769a625025954a7eadff8c.jpg	640.0	373.0
32231	7248188.0	6814336_e0816623a59b1c64657e8eead09bdb02.jpg	640.0	427.0
3467	6814336.0	6814336_e0816623a59b1c64657e8eead09bdb02.jpg	640.0	427.0
32232	7248188.0	6814336_e76edd169b8581b047352649185c2b78.jpg	640.0	480.0

- Features extracted from single listing may not be uniformly distributed.

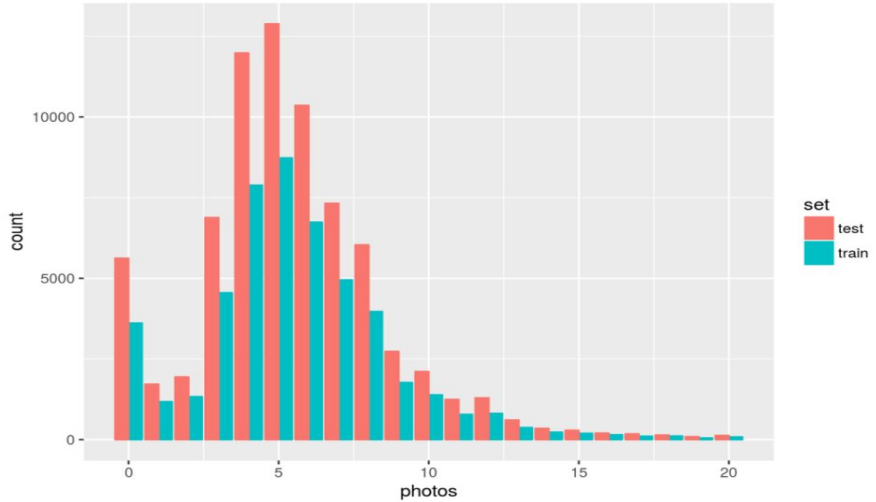
Solutions:

- Compressed Images

  100*100 size.

  100*100 thumbnail takes about 4k size

  Total size = total photos * size of each photo

  2.7 GB Memory Space
- Used NA values as 0, duplicates were left untouched.
- 

  Images in Test and Train data were uniformly distributed.

Features Extraction

- Extraction of 15 labels from each images:

photos.renthop.com/2/7211212_1ed4542ec...	indoors	family	room	contemporary	furniture	house	trading floor	door	apartment	interior design	empty	inside	window
photos.renthop.com/2/7150865_be3306c5d...	indoors	window	empty	room	no person	family	interior design	house	furniture	contemporary	wall	wood	home
photos.renthop.com/2/6887163_de85c4273...	indoors	room	house	family	furniture	trading floor	door	empty	apartment	inside	contemporary	interior design	ceiling
photos.renthop.com/2/6888711_6e660cee4f...	indoors	room	contemporary	family	furniture	house	apartment	window	empty	inside	interior design	trading floor	ceiling
photos.renthop.com/2/6934781_1fa4b41a92...	bathroom	contemporary	indoors	washcloset	room	bathtub	no person	family	clean	inside	empty	luxury	interior
photos.renthop.com/2/6894514_9abb85920...	city	road	architecture	building	street	travel	modern	downtown	traffic	urban	car	transportation system	busines
photos.renthop.com/2/6930771_7e3622b6a...	furniture	room	indoors	sofa	contemporary	window	seat	interior design	apartment	rug	house	trading floor	chair
photos.renthop.com/2/6867392_b18283f6bc...	room	indoors	table	inside	family	furniture	contemporary	modern	stainless steel	house	stove	chair	luxury
photos.renthop.com/2/6898799_3759be4c8...	room	furniture	sofa	indoors	house	contemporary	interior design	apartment	seat	lamp	family	inside	trading
photos.renthop.com/2/6814332_e19a8552bf...	furniture	table	room	chair	window	indoors	seat	contemporary	interior design	apartment	trading floor	dining room	house
photos.renthop.com/2/6869199_06b2601f05...	bathroom	room	contemporary	indoors	furniture	faucet	family	interior design	luxury	bathtub	no person	inside	shower
photos.renthop.com/2/7102986_ca6af538c2...	indoors	ceiling	inside	dug-out pool	lobby	airport	subway system	public	train	trading floor	swimming pool	modern	escalat
photos.renthop.com/2/6895442_34d617a5f3...	architecture	city	building	modern	daylight	office	sky	urban	no person	business	travel	outdoors	constru
photos.renthop.com/2/6846213_832587546...	chair	seat	no person	luxury	furniture	architecture	travel	garden	bench	outdoors	street	house	empty
photos.renthop.com/2/7089402_66038eaa98...	indoors	grinder	room	no person	industry	production	business	energy	stainless steel	equipment	tube	distillery	steel
photos.renthop.com/2/6889043_a3e1c00433...	dug-out pool	hotel	leisure	recreation	no person	playground	water	house	resort	color	travel	vacation	family
photos.renthop.com/2/6913348_829f19ac3c...	indoors	interior design	furniture	contemporary	room	no person	home	chair	window	seat	sofa	family	table
photos.renthop.com/2/6894111_571fa57d00...	indoors	contemporary	furniture	room	interior design	family	comfort	sofa	apartment	parquet	wood	luxury	house
	no person	architecture	outdoors	house	travel	horizontal plane	city	family	apartment	street	luxury	modern	home
photos.renthop.com/2/6848536_9a3e1f8778...	indoors	no person	contemporary	family	house	outdoors	bathroom	room	modern	window	clean	architecture	luxury
photos.renthop.com/2/6858062_5cfb9d9002...	indoors	window	no person	interior design	contemporary	furniture	home	house	room	door	family	glass items	architec

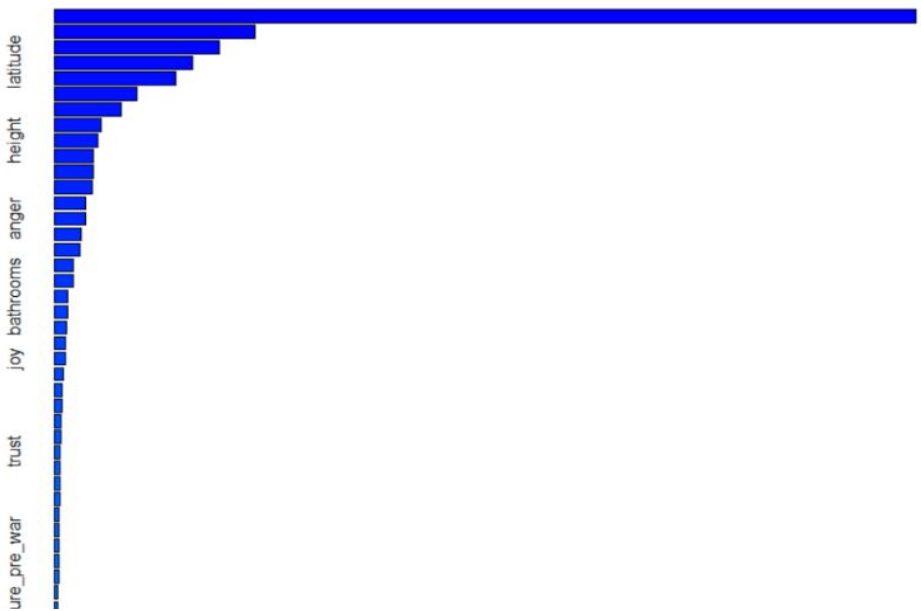
Features Extraction

- Image width and Height and levels
- Blur Detection

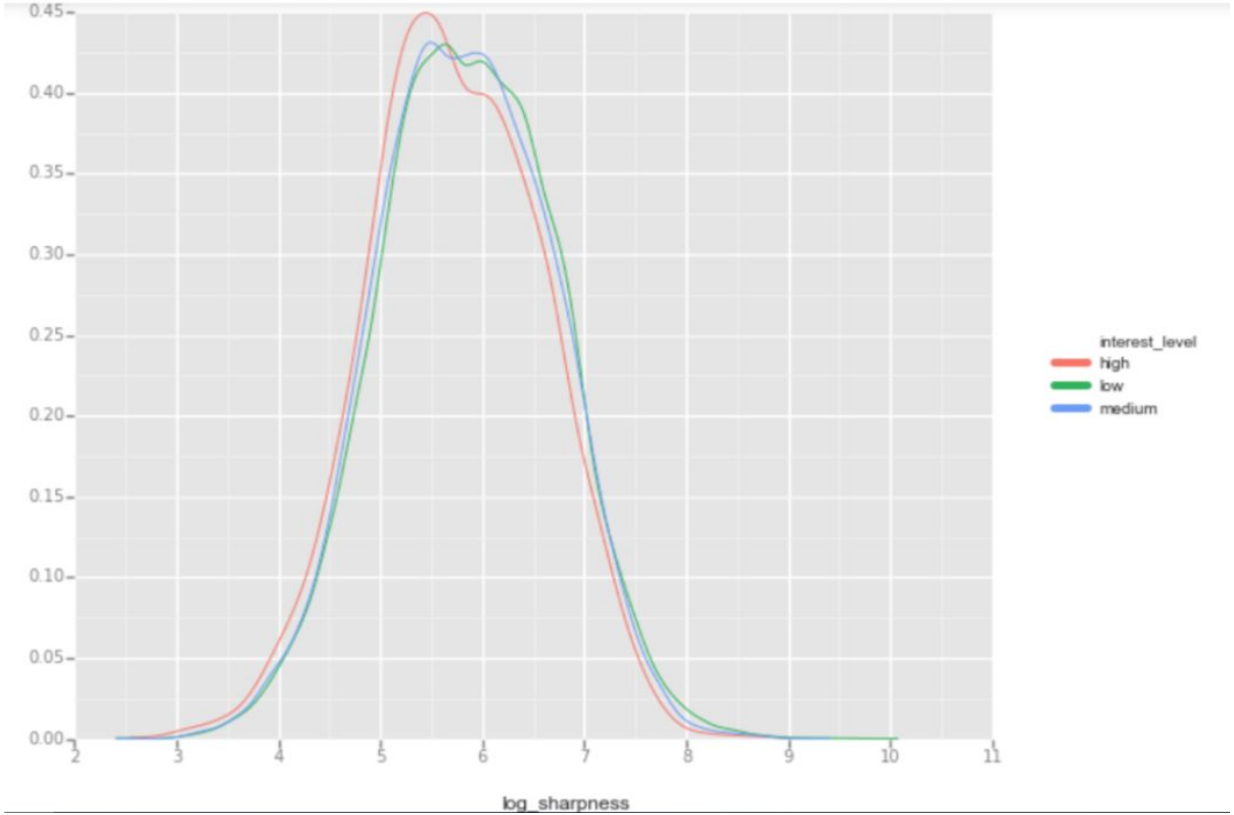
listing_id	photo_name	width	height	layers	sharpness
6811957	6811957_33d08c8dc440c89bccc8d9889c5485a6.jpg	640	426	3	85.55963
6811957	6811957_3dad56e8bf3477b2900ca39d57df041e.jpg	640	426	3	175.55901
6811957	6811957_7d3ab8175d23fee64c0651b1bc16f2cc.jpg	426	640	3	85.44322
6811957	6811957_83a4e2e75ea15a5a1d2cc9de3407b1b9.jpg	640	426	3	62.21313
6811957	6811957_acbdbbe6ff435b9d4f520db6da1ada9a.jpg	640	426	3	132.29719
6811958	6811958_1fe0076c8b481e0af2223afab02503da.jpg	640	425	3	365.98526
6811958	6811958_23ceae42d2c88ffc5a057db1deb346c1.jpg	640	426	3	663.17997
6811958	6811958_bb863a4184a1e085f0c55e0172767abd.jpg	344	544	3	6215.06184
6811958	6811958_c131c57b97afd739161579cdb41c9884.jpg	640	425	3	468.23858

Infereces: Photo Importnce

var	rel.inf
price	3.939148e+01
bedrooms	9.158772e+00
longitude	7.536486e+00
latitude	6.309691e+00
feature_hardwood_floors	5.571126e+00
num_description_words	3.770432e+00
X	3.073653e+00
log_sharpness	2.121494e+00
height	1.989015e+00
num_photos	1.807549e+00
width	1.805829e+00
feature_laundry	1.733271e+00
created_day	1.448742e+00
anger	1.434664e+00
feature_dishwasher	1.212838e+00
num_features	1.194198e+00
feature_simplex	8.720697e-01
feature_no_fee	8.617346e-01
bathrooms	6.165504e-01
feature_short_term_allowed	5.957912e-01



Inference: Sharpness on Interst Level



The orange color shows high interest level

The x axis shows sharpness

The listing with blur photos are

Showing higher interest level

Throughout the training listings.

Natural Language Processing (NLP)

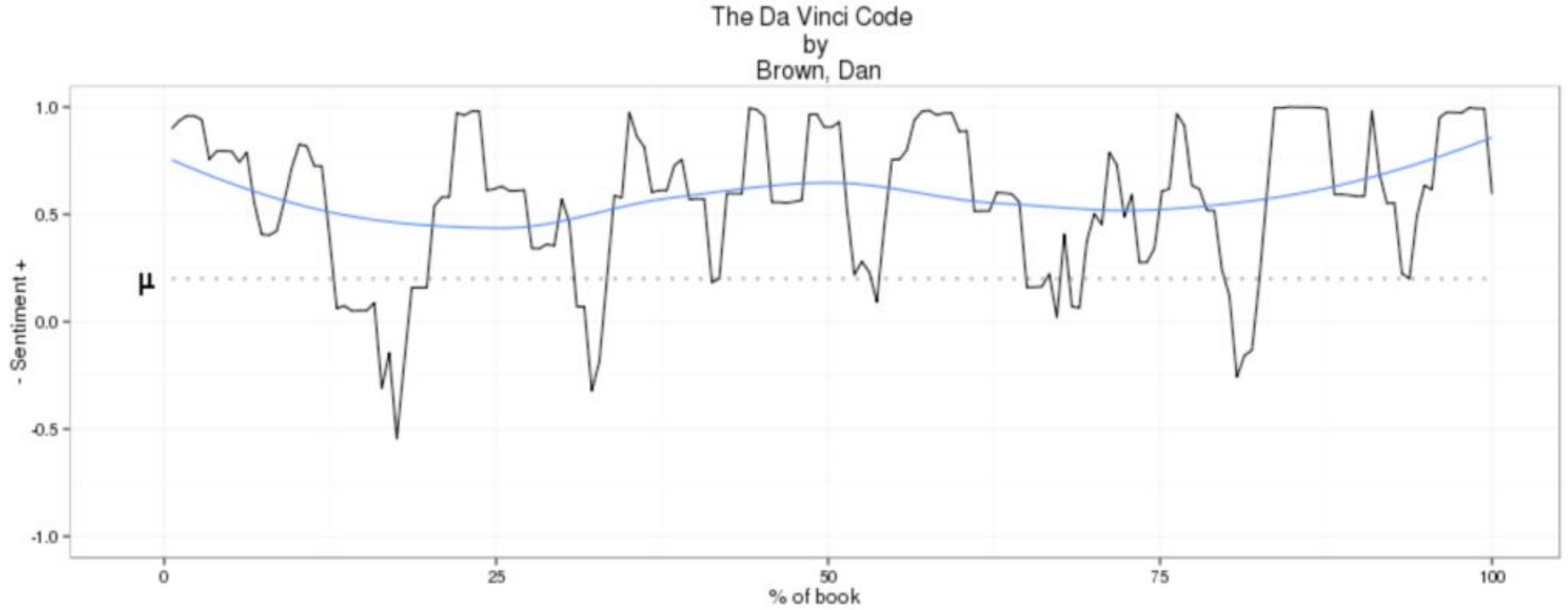
An R package for the extraction of sentiment and sentiment-based plot arcs from text.

The name "Syuzhet" comes from the Russian Formalists Victor Shklovsky and Vladimir Propp who divided narrative into two components, the "fabula" and the "syuzhet." Syuzhet refers to the "device" or technique of a narrative whereas fabula is the chronological order of events. Syuzhet, therefore, is concerned with the manner in which the elements of the story (fabula) are organized (syuzhet).

The Syuzhet package attempts to reveal the latent structure of narrative by means of sentiment analysis. Instead of detecting shifts in the topic or subject matter of the narrative (as Ben Schmidt has done), the Syuzhet package reveals the emotional shifts that serve as proxies for the narrative movement between conflict and conflict resolution.

- Afinn - develoepd by Finn Arup Nielsen as the AFINN WORD DATABASE
- Bing - develoepd by Minqing Hu and Bing Liu as the OPINION LEXICON
- NRC - developed by Mohammad, Saif M. and Turney, Peter D. as the NRC EMOTION LEXICON

Sentiment Analysis of a Good Novel



What is Sentiment Analysis?

Opinion mining or sentiment analysis Computational study of opinions, sentiments, subjectivity, evaluations, attitudes, appraisal, affects, views, emotions, etc., expressed in text. Reviews, blogs, discussions, news, comments, feedback, or any other documents

“Opinions” are key influencers of our behaviors. Our beliefs and perceptions of reality are conditioned on how others see the world. Often when we need to make a decision, we often seek out the opinions of others. In the past, Individuals: seek opinions from friends and family Organizations: use surveys, focus groups, opinion polls, consultants.

Structured the Unstructured

Syuzhet is concerned with the linear progression of narrative from beginning (first page) to the end (last page), whereas *fabula* is concerned with the specific events of a story, events which may or may not be related in chronological order … When we study the *syuzhet*, we are not so much concerned with the order of the fictional events but specifically interested in the manner in which the author presents those events to readers.

Using NLP for Feature Extraction

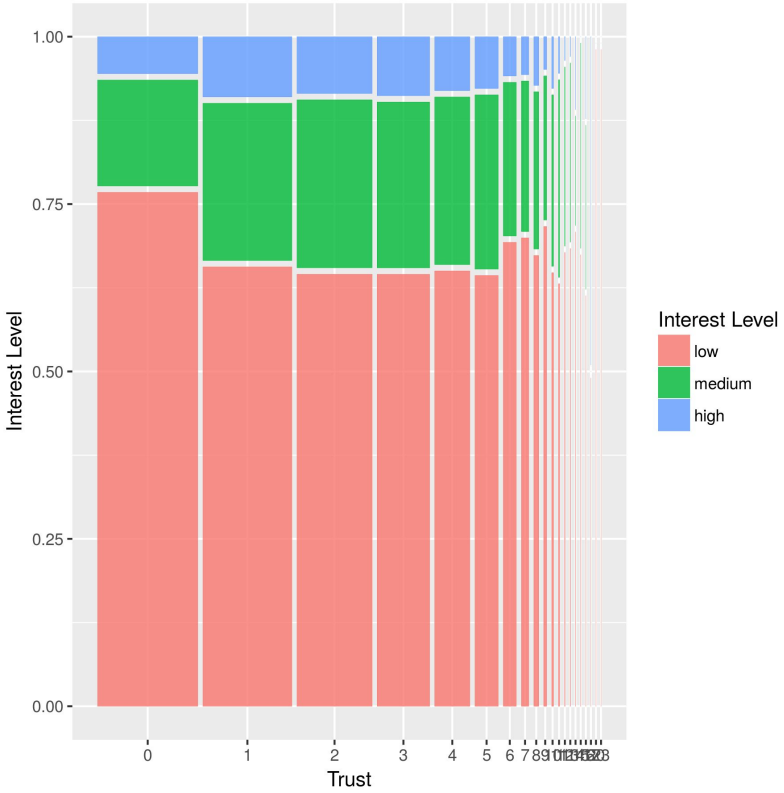
We used the syuzhet package to take the descriptions from the Kaggle dataset and produce a scoring mechanism that valued each description on a range of emotions:

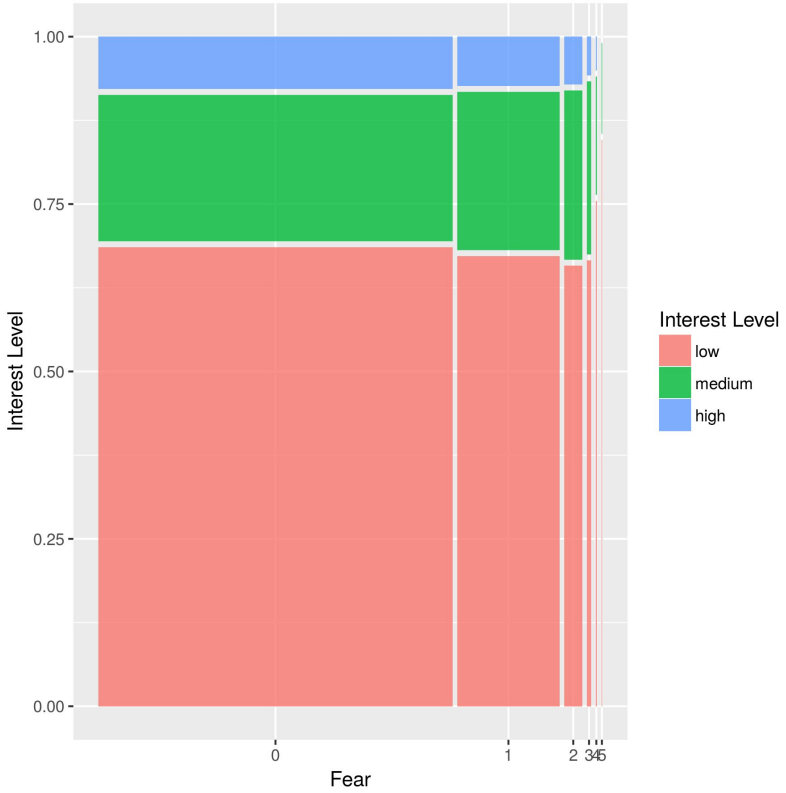
'Anger','anticipation','disgust','fear','joy','sadness','surprise','trust',

In addition we composed additional features for these datasets based on valence as being either:

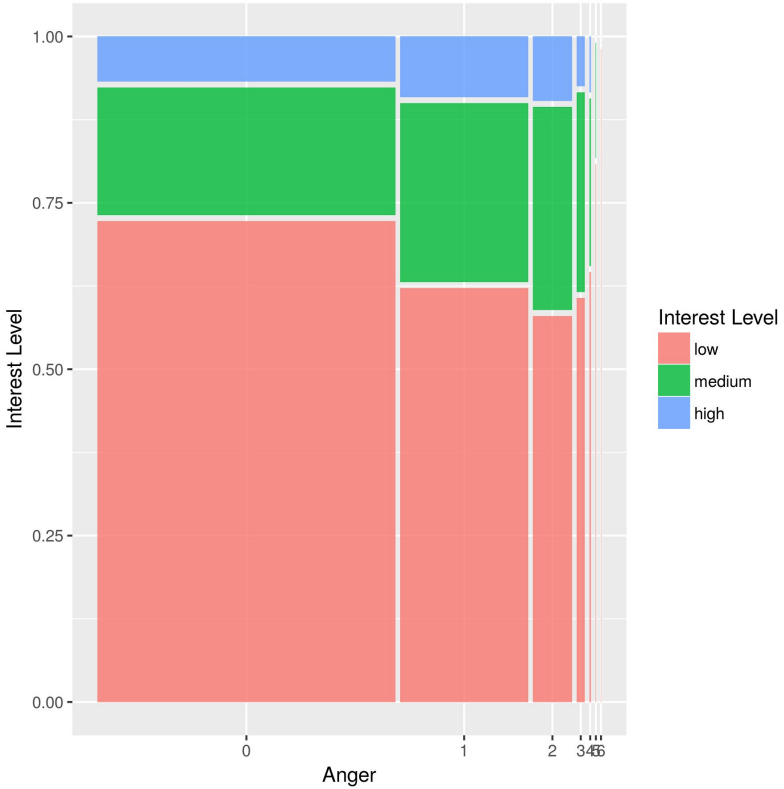
'negative','positive'

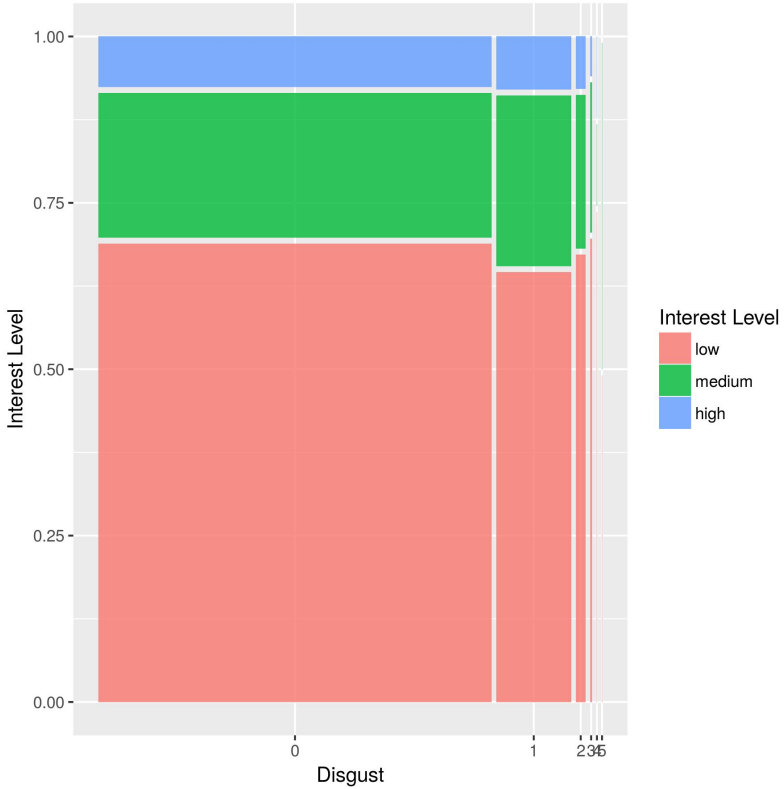
Trust - Fear


1.00
0.75
0.50
0.25
0.00
Interest Level
0
1
2
3
4
5
6
Trust
Interest Level
low
medium
high

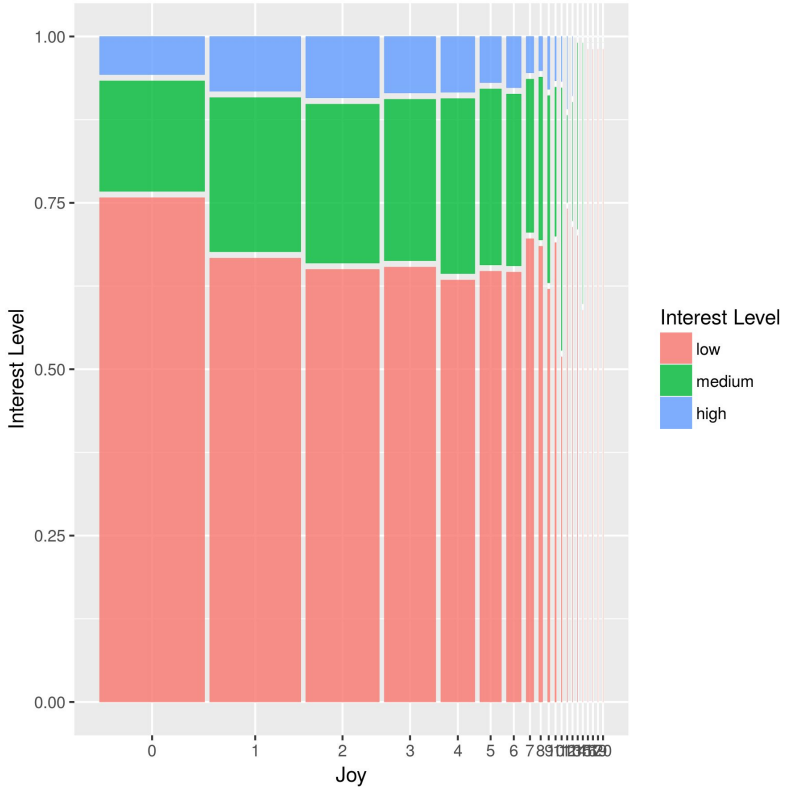

1.00
0.75
0.50
0.25
0.00
Interest Level
0
1
2
3
4
5
Fear
Interest Level
low
medium
high

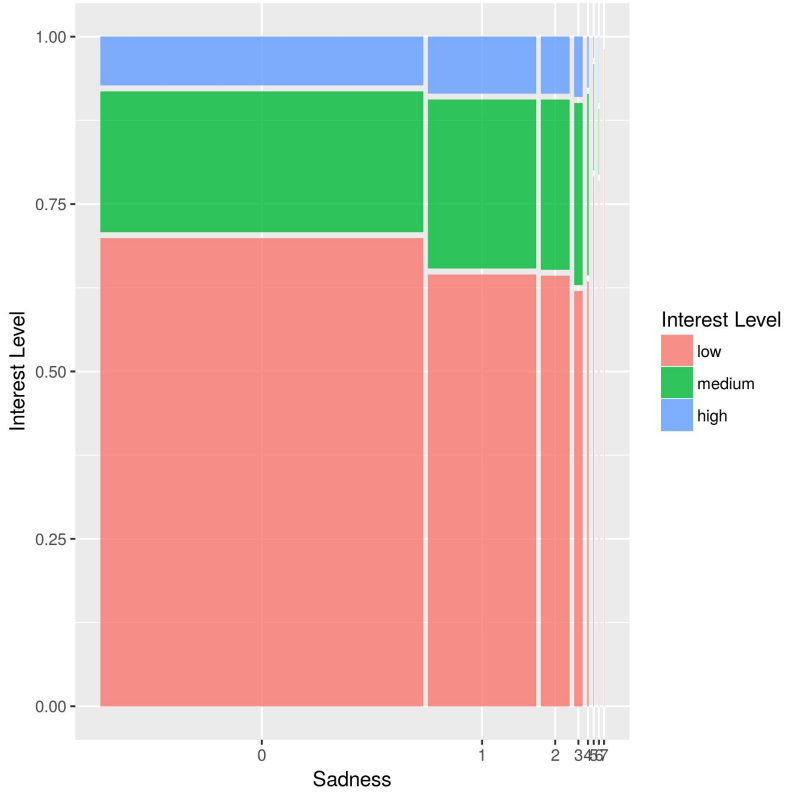
Anger - Disgust



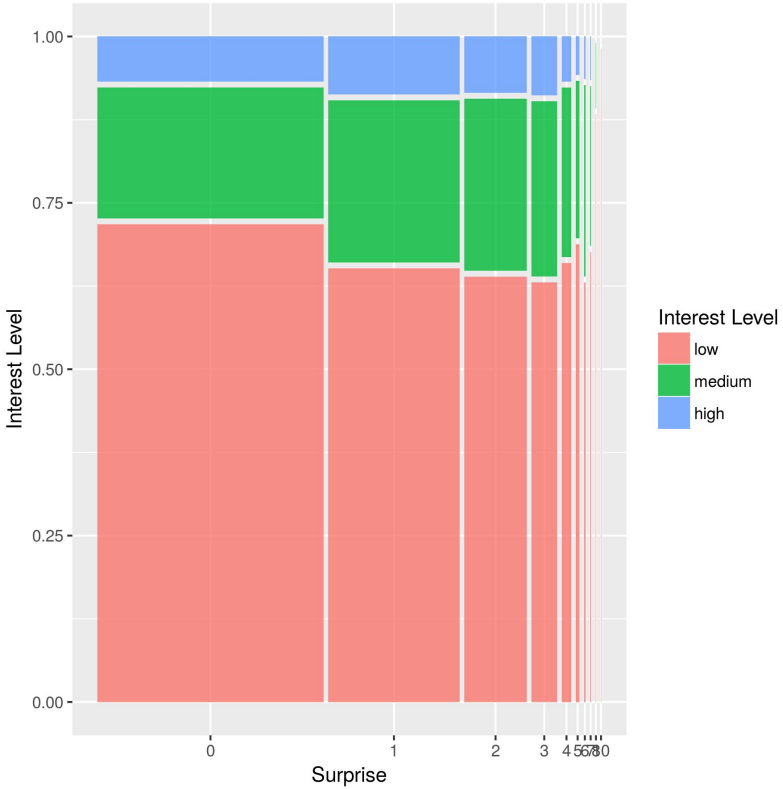


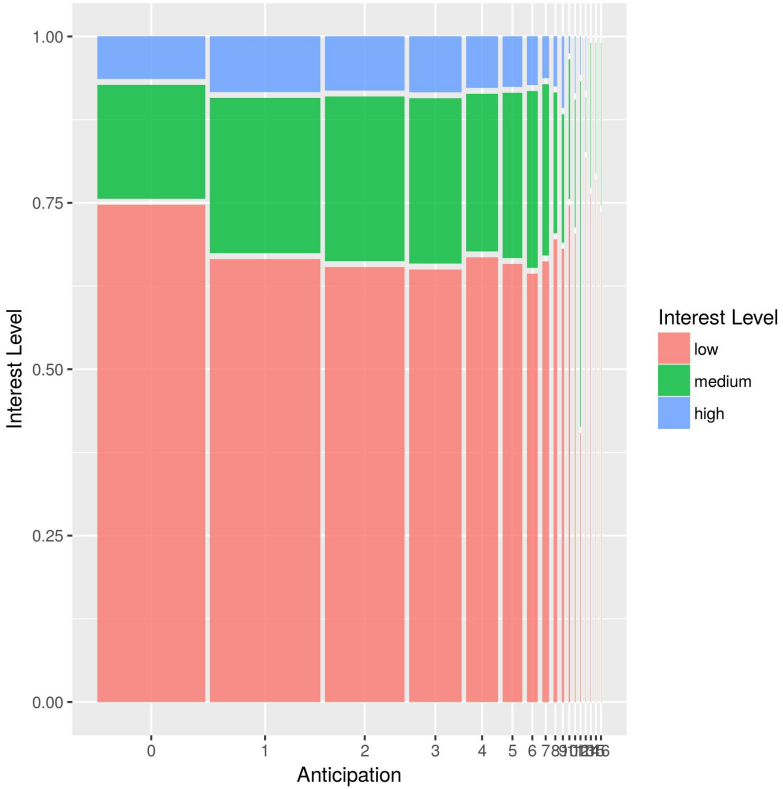
Joy - Sadness


1.00
0.75
0.50
0.25
0.00
Interest Level
0
1
2
3
4
5
6
Joy
Interest Level
low
medium
high

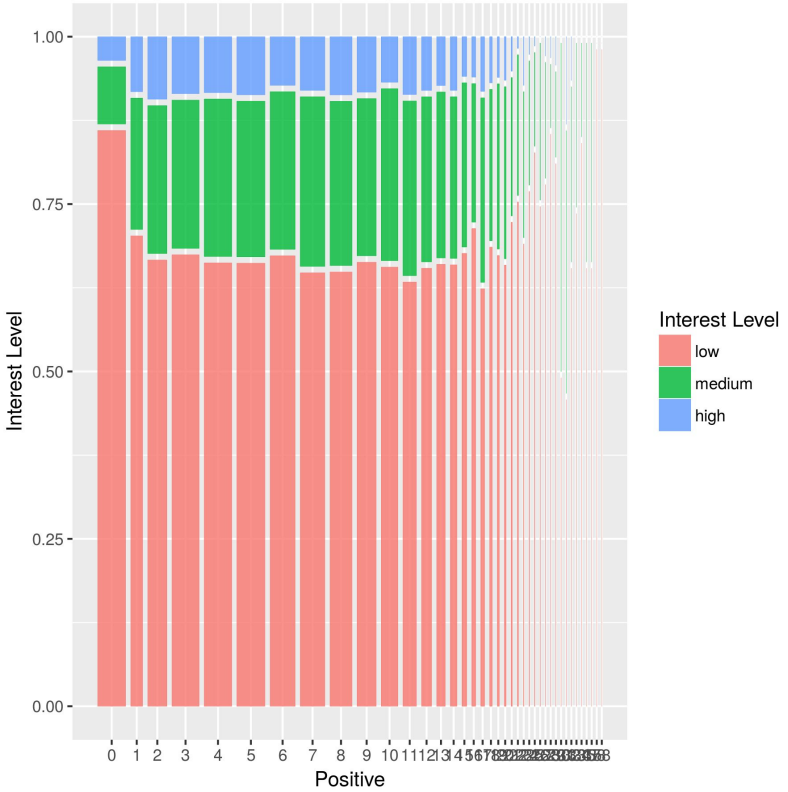

1.00
0.75
0.50
0.25
0.00
Interest Level
0
1
2
Sadness
Interest Level
low
medium
high

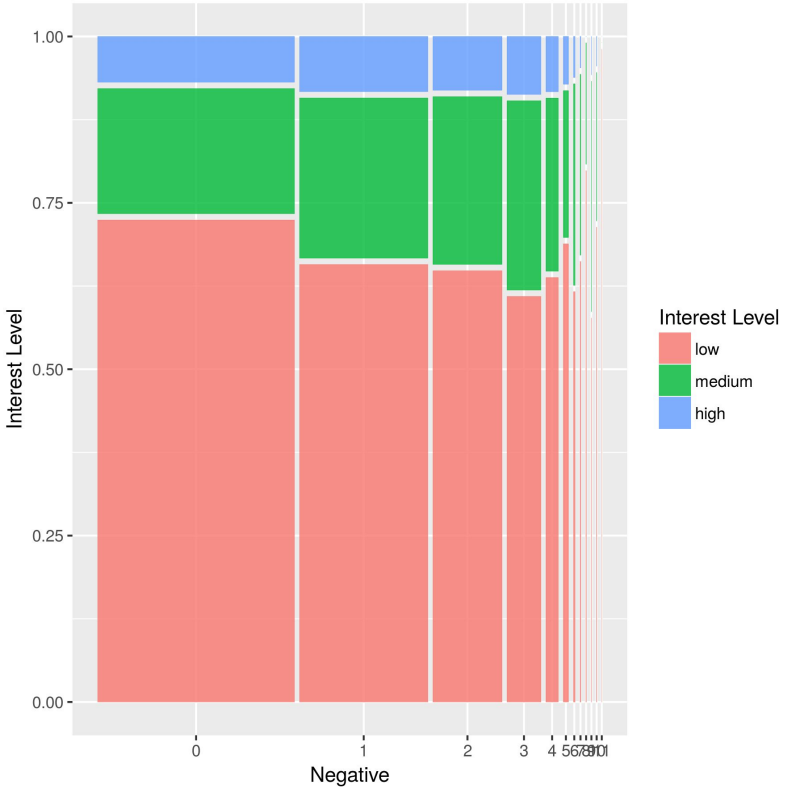
Surprise - Anticipation





Positive vs. Negative





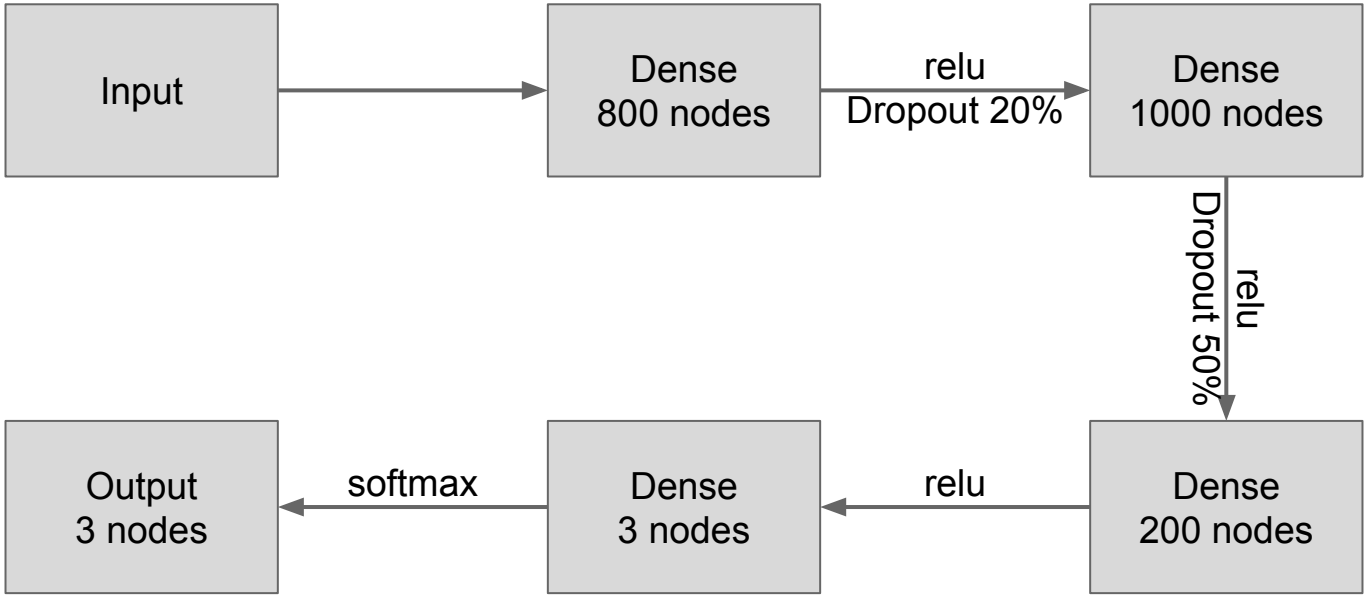
Neural Network Approach

- Most Kaggle Kernels focused on Decision Tree based models
- More than 99% of the data is contained in the photos.

How to take advantage of it?

- Feature engineering: extract some *statistics*, such as dimensions, brightness, sharpness/blur, etc
- Train a separate (convolutional neural network) model that classifies each image (possible categories: kitchen, bathroom, garden, street view, gym)
- Extract *anonymous features* by training in a unified model

Main NN Model Structure

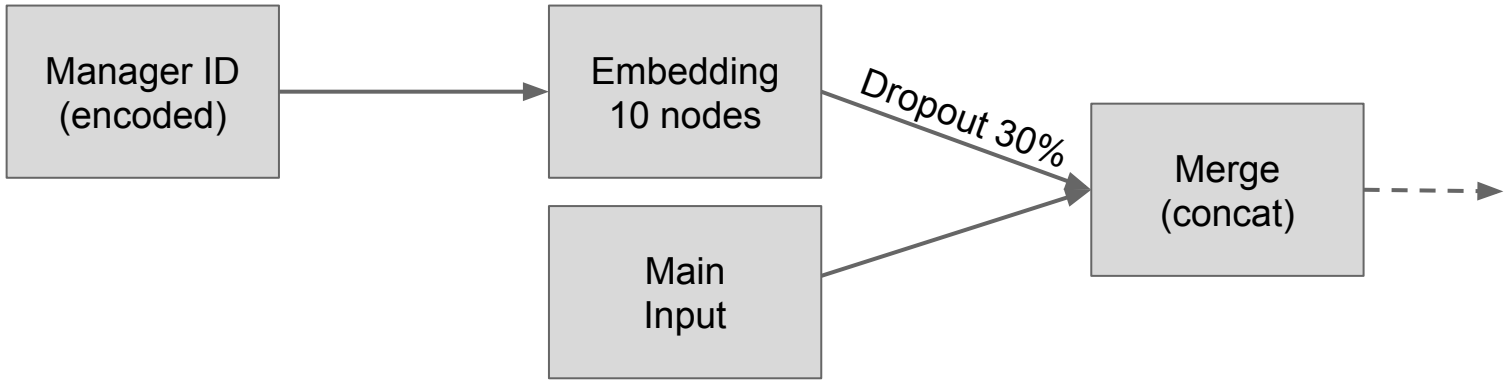


Input

- Original numerical variables (coordinates *snapped*, price in log scale)
- New variables: price per bedroom/bathroom (log), number of photos, number/word count of features, length/word count of description, hour of day (fractional), day of week, day of month, month
- 56 distinct apartment features as dummy variables (regex extracted)
- Output of sentiment analysis on description
- Manager ID: how to do it?...

Manager ID

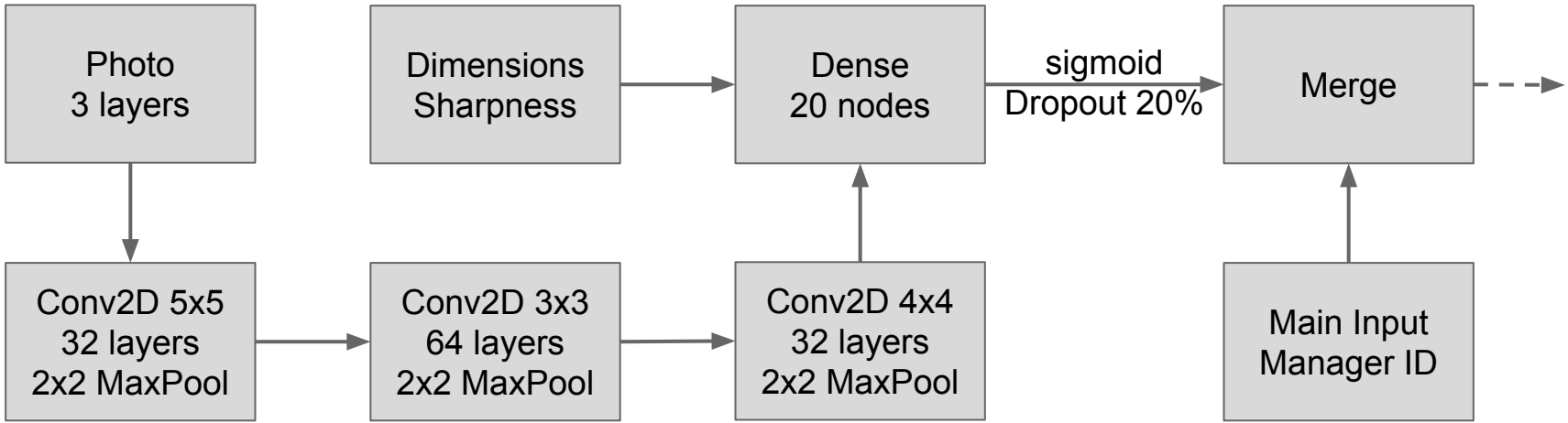
We encoded the top 999 manager ids based on the number of listings on the training data to integers from 1 to 999, mapping the remaining ones to 0.



Embedding: 1000x10 trainable values, k-th manager mapped to k-th row

What about the photos?

- Resized to 100x100 thumbnails (arrrgh)
- Original dimensions and sharpness extracted from originals
- System constraints discourage loading more than 20k photos simultaneously



Results from using photos

- Training takes much longer: 2 mins/epoch for 20k listings
- Deeper network: harder to tune and train
- Approach taken: train until early stopping, extract weights from convolutional part, compute activations for all listings, train on all listings with activations as extra input.
- Kaggle score: 0.59709. Worse than without photos: 0.58854
- Takeaways: try some tuning, train a model exclusively on photos, or include activations from convolutional branch later in the model (not side-by-side with inputs)

NN implementation

```
def features_sentiment_manager_sharpness_preprocessor():
    json_loader = loaders.JSONLoader()
    preprocessor = loaders.Preprocessor()
    preprocessor.with_pipeline('origin').set_loader(json_loader)
    preprocessor.add_operation(loaders.DateTimeExtractor()).add_operation(loaders.NewSimplePredictors
    preprocessor.add_operation(loaders.LogTransform(['price_per_bedroom', 'price', 'price_per_bathroo
    preprocessor.add_operation(loaders.Selector(['listing_id', 'bathrooms', u'bedrooms', 'latitude',
                                                 'month', 'day_of_month', 'hour', 'day_of_week', 'pri
                                                 'price_per_bedroom', 'num_features', 'features_len',
    merger = loaders.PandasColumnMerger(['origin', 'features', 'sentiment', 'photo_stats'], on = 'lis
    preprocessor.set_consumer(merger)
    preprocessor.with_pipeline('features').set_loader(loaders.CSVLoader('data/features_train.csv', 'd
    preprocessor.set_consumer(merger)
    preprocessor.with_pipeline('sentiment').set_loader(loaders.CSVLoader('data/sentiment_train.csv',
    preprocessor.set_consumer(merger)
    photo_url_merger = loaders.GetTopPhotoMerger('photo_stats_sharpness', 'photo_stats_photo_url')
    preprocessor.with_pipeline('photo_stats_sharpness').set_loader(loaders.CSVLoader('data/images_tra
    preprocessor.set_consumer(photo_url_merger)
    preprocessor.with_pipeline('photo_stats_photo_url').set_loader(json_loader.select_loader(['listin
    preprocessor.set_consumer(photo_url_merger)
    preprocessor.with_pipeline('photo_stats').set_loader(photo_url_merger).add_operation(loaders.Colu
    preprocessor.add_operation(loaders.LogTransform(['avg_sharpness', 'cover_sharpness'])).set_consum
    preprocessor.with_pipeline('main').set_loader(merger).add_operation(loaders.ColumnDrop('listing_i
    preprocessor.add_operation(loaders.ToNdarray()).add_operation(preprocessing.StandardScaler())
    preprocessor.with_pipeline('response').set_loader(json_loader.select_loader('interest_level'), on
    preprocessor.add_operation(loaders.Dummifier(output_cols = ['high', 'medium', 'low'])).add_operat
    preprocessor.with_pipeline('managers').set_loader(json_loader.select_loader('manager_id'))
    preprocessor.add_operation(loaders.CategoricalFilter(999)).add_operation(loaders.ToNdarray(dtype
    preprocessor.with_pipeline('ids').set_loader(json_loader.select_loader('listing_id'))
    preprocessor.add_operation(loaders.ToNdarray(dtype = np.int64))
    return preprocessor
```

- Keras framework with Theano backend
- Built a dedicated pipeline framework for feature engineering

Key Takeaways

- Machine learning is fun and it is surprisingly easy to get your hands wet
- Use sparse data formats wherever possible and do not mix up pandas sparse with scipy sparse
- Build your environment and take it wherever you go
- Cloud computing is fast but learning to use it is slow
- Test extensively and fully debug before executing on IaaS

Questions?