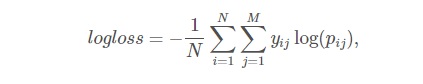
**Methodology**

For Two - Sigma Rent Submissions are evaluated using the [multi-class logarithmic loss](https://www.kaggle.com/wiki/MultiClassLogLoss). Each listing has one true class. For each listing, we must submit a set of predicted probabilities one for every listings.

The formula is then,



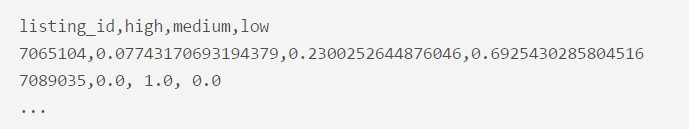
where N is the number of listings in the test set, M is the number of class labels 3classes, log is the natural logarithm, yij is 1 if observation, i belongs to class, j and 0 otherwise, and pij is the predicted probability that observation.

The submitted probabilities for a given listing are not required to sum to one because they are rescaled prior to being scored each row is divided by the row sum. In order to avoid the extremes of the log function, predicted probabilities are replaced with max(min(p,1−10−15,10^{-15})).

**Submission File Criteria:**

For this competition we have submited a csv file with the listing\_id, and a probability for each class.

The order of the rows does not matter. The file must have a header and it should look like the following:



**Data Provided**

* **train.json -** the training set
* **test.json -** the test set
* **sample\_submission.csv -** a sample submission file in the correct format
* **images\_sample.zip -** listing images organized by listing\_id (a sample of 100 listings)
* **Kaggle-renthop.7z**- (optional) listing images organized by listing\_id. Total size: 78.5GB compressed.

**Data Variables to project output**

|  |  |
| --- | --- |
| * **Bathrooms:** Number of bathrooms * **Bedrooms:** Number of bedrooms * **Building Id** * **Created:** Date of created * **Description** * **Display Address:** Address of an apartment * **Features:** Features about thisapartment * **latitude** * **Longitude** | * **Listing Id** * **Manager Id** * **Photos**: a list of photo links. You are welcome to download the pictures yourselves from RentHop's site, but they are the same as imgs.zip. * **Price:** In USD * **Street Address** * **Interest Level:** This is the target variable. It has 3 categories: 'high', 'medium', 'low' |

**Challenges and Solutions**

### For Two sigma connect: rental listing inquiries competition, we have started our prediction by using R programming machine learning algorithms. We have faced first challenge as the data provided was in JSON format. So, first we converted that data into regular format by using JSONLITE library. After that we have used the Purr library to remove the duplicated data. Next Challenge we have faced that we have to submit an output in multinomial format i.e Listing Id (Low, Medium, High). For that we have tried Quadratic Discriminant Analysis (QDA) algorithm, Random Forest Algorithm. Which all of these algorithm could not help to predict multinomial data. So, finally we ended up using Multinomial Logistic Regression with NNET library.

**Coding**

#**train dataset json to csv**

train <- fromJSON("train.json")

vartrain <- setdiff(names(train), c("photos", "features"))

train <- map\_at(train, vartrain, unlist) %>% tibble::as\_tibble(.)

colnames(train)

train = train[, -12]

#**test dataset json to csv**

test <- fromJSON("test.json")

vartest <- setdiff(names(test), c("photos", "features"))

test <- map\_at(test, vartest, unlist) %>% tibble::as\_tibble(.)

colnames(test)

test = test[, -12]

**#convert into factors**

train$interest\_level = factor(train$interest\_level, levels = c('low', 'medium', 'high'))

**# model creation**

model = multinom(interest\_level ~ price + bathrooms + bedrooms ,data=train)

**# test prediction**

probs <- predict(model, test, "probs")

test\_df = data.frame(listing\_id = test$listing\_id, probs)

write.csv(test\_df, file = "submission.csv", row.names = FALSE)

**Output**

listing\_id low medium high

1 7142618 0.7126286 0.2234512 0.06392011

2 7210040 0.5782710 0.3063397 0.11538935

3 7174566 0.7216703 0.2068264 0.07150331

4 7191391 0.5872913 0.3022935 0.11041518

5 7171695 0.7543269 0.1985634 0.04710969

6 7225206 0.8046366 0.1735269 0.02183648