**Predicting the Uncertainty of the Sales**

Amit Singh

Department of Electrical Engineering

Stevens Institute of Technology Hoboken, NJ

[asingh61@stevens.edu](mailto:asingh61@stevens.edu)

Ganapathi Ekambaram

Department of Electrical Engineering

Stevens Institute of Technology Hoboken, NJ

gekambar[@stevens.edu](mailto:asingh61@stevens.edu)

Abstract—The goal of this study is to use hierarchical sales data from Walmart, the world’s largest company by revenue, to forecast daily sales for the next 28 days and to make uncertainty estimates for these forecasts. The data covers stores in three US States (California, Texas, and Wisconsin) and includes item level, department, product categories, and store details. In addition, it has explanatory variables such as price, promotions, day of the week, and special events. Several machine learning techniques were applied to problem, trained on a wide range of features, with varying success.

Index Terms—Artificial Neural Network, Linear Regression, LSTM, Time Series Prediction, Feature Extraction, Supervised Learning, Gaussian process, Uncertainty in Sales

1. **INTRODUCTION**

Predicting the sales of an organization play a critical role in the longevity of an organization. It may seem like a daunting task as predicting the weather, but both rely on the science and historical data. The sales data, store id, product code, location, time of the sales etc. holds a key to predict the future of sales. Understand the data and figure out the features which have more weightage and correlation in the sales. Is there a pattern in the sales? Once we have figured out a scientific way- machine learning way to predict the sales, what is the uncertainty of that prediction, estimate the levels of uncertainty. We have tried to look at uncertainty as a Bayesian process and modulate sales as an interval of uncertainty.

To accomplish, we have pulled the dataset from the Kaggle competition.

II **TOOLS USED**

* Sklearn
* Keras
* TensorFlow
* Matplotlib
* Numpy
* Pandas
* JuypterNOTEBOOK/Anaconda
* Kaggle
* Statsmodels

## **III Git Repository**

<https://github.com/as-stevens/m5sales-uncertanity>

**IV DATA COLLECTION**

The data for this project is collected from the Kaggle competition; <https://www.kaggle.com/c/m5-forecasting-uncertainty/data>

The data consisted of the following;

* Calendar.csv- This contains the calendar data and has the columns;
  + Date – the date
  + wm\_yr\_wk – week month of the year
  + weekday – Saturday or Sunday
  + wday – week of the day 1 to 7
  + month – month of the year
  + year – the year
  + d – an id for the date in d\_11 format
  + event\_name\_1 – weather the date is any event date
  + event\_type\_1 - weather the date is any event date
  + event\_name\_2 - weather the date is any event date
  + event\_type\_2 - weather the date is any event date
  + snap\_CA – snap data for CA
  + snap\_TX - snap data for TX
  + snap\_WI - snap data for WI
* Sales\_train\_validation.csv - Contains the historical daily unit sales data per product and store [d\_1 - d\_1913]
  + Id – Unique record for the row item
  + item\_id - Item id
  + dept\_id – department id
  + cat\_id – category id of the product
  + store\_id - store id
  + state\_id – State id
  + d\_1 - d\_1913 – day represented as d\_1 to d\_1913
* Sell\_price.csv - Contains information about the price of the products sold per store and date.
  + store\_id
  + item\_id
  + wm\_yr\_wk
  + sell\_price

## **V Related Work**

This project is divided into two sections, sales prediction and uncertainty to sales prediction, **Ganapathi Ekambaram** has worked on **Sales Prediction** and **Amit Singh** has worked on **Uncertainty**.

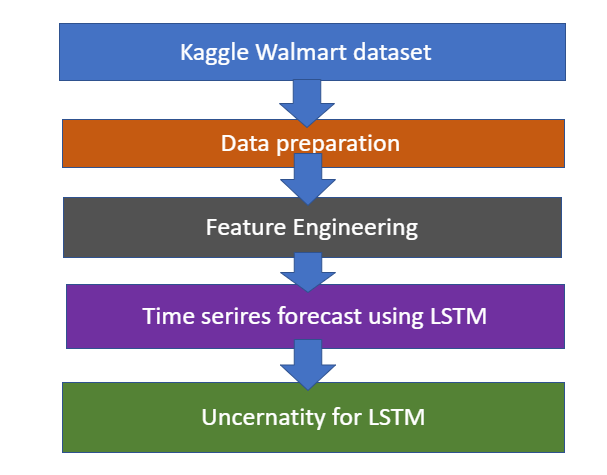
There are two parallel competitions: Accuracy and Uncertainty, we are trying to predict sales for 28 forecast days for specific item from sales data of state, store, department and item level sales data for 5 years

The accuracy competition will use the metric: **Weighted Root Mean Squared Scaled Error (RMSEE)**

The uncertainty competition will use the metric: **Weighted Scaled Pinball Loss (WSPL)**

### **Project overview**

We have used Kaggle public data source to predict sales forecast and uncertainty, statistical data analysis has performed to extract feature from given data files, discussed numerous supervised and unsupervised algorithms before we have opted RNN for time series prediction, developed Bayesian likelihood for Neural network dropout to predict uncernity.

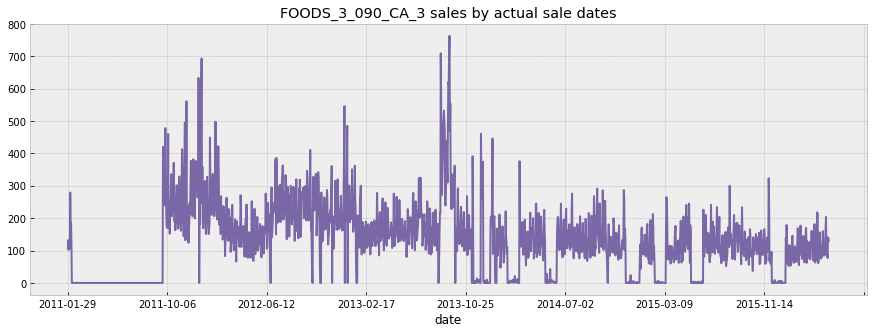


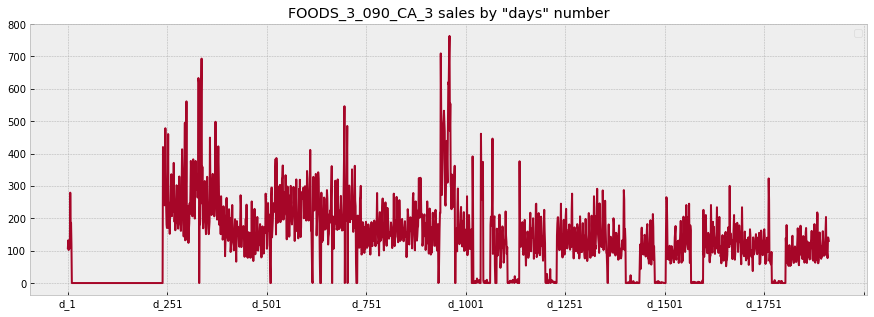
### **Data Preparation**

The hierarchical data covers three stores data for 3 years, it includes store, category, department and item level. In addition, data is explanatory such as price, promotions, day of the week and special events.

Try a random item that sells a lot and see how it’s sales look across the training databased on chart, there are days when item not available

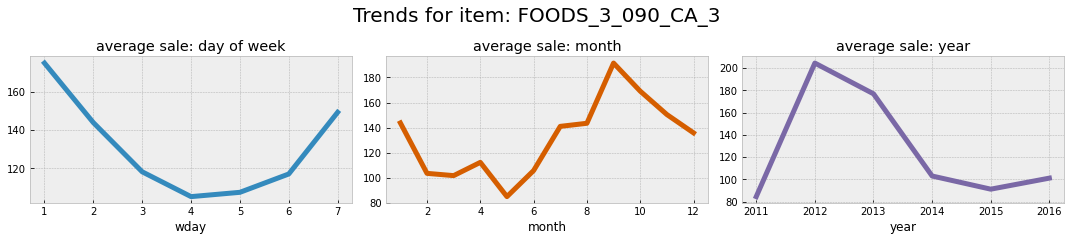
Data for single item





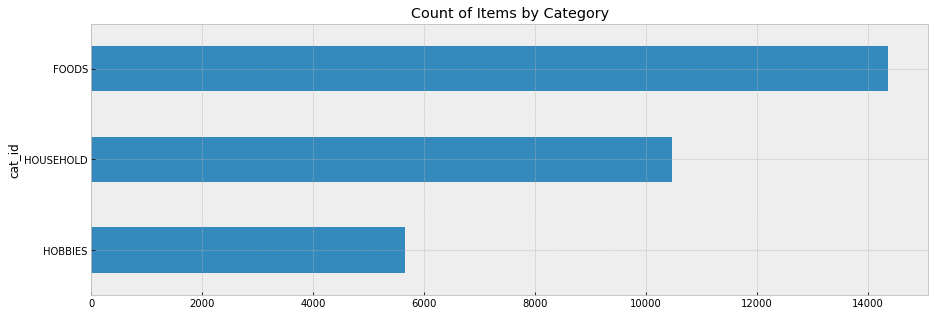
Sales broken down by time

Try to analyze how particular item sells in week day, month and year



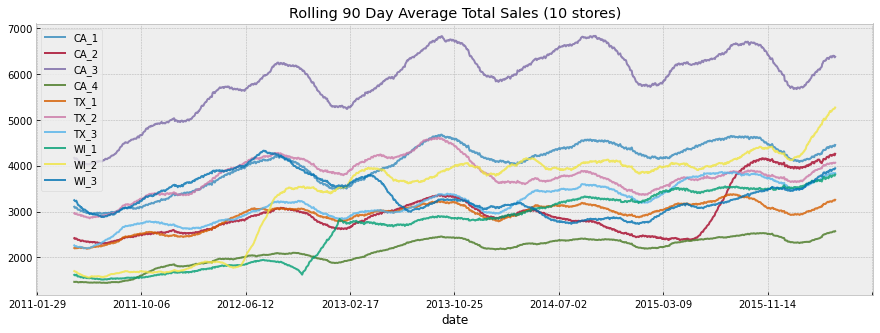
Combined Sales Over Time by Type

Combination of Hobbies, household and food by time frame shown here.

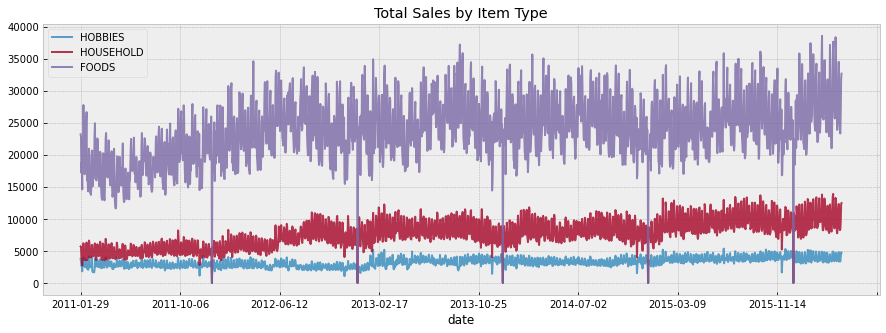


Sales by Store

10 unique items what is sold in a store



Sales by Item Type

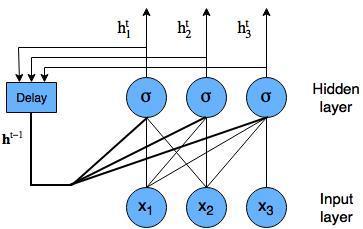


## **Sales Prediction**

Model

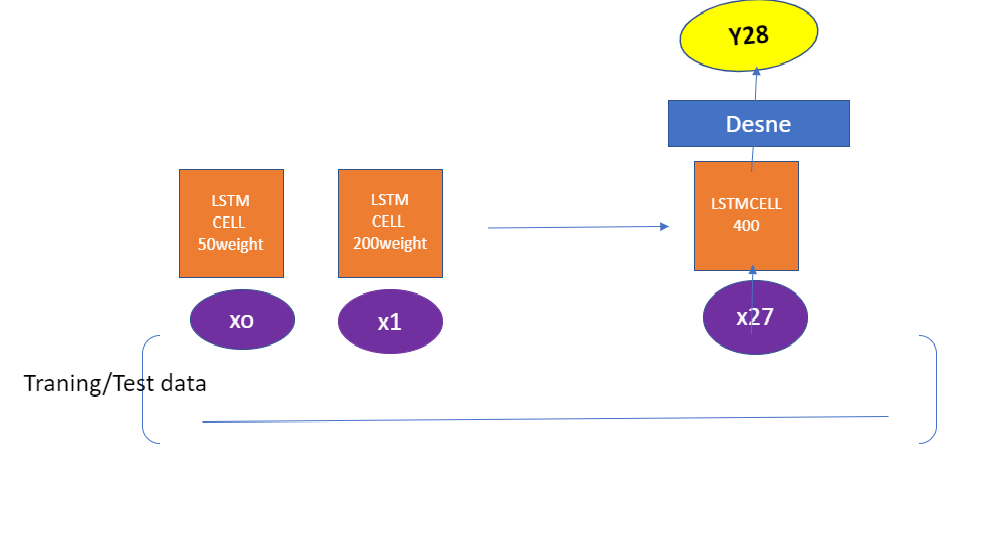
We have evaluated various supervised and unsupervised algorithms for sales predictions, regression and decision tree algorithms have been tried, in order to predict uncertainty using Bayesian inference, we have decided to build neural network model, LSTM layers is best for time series analysis.

Long Short-Term Memory modal is a recurrent neutral network architecture used in deep earning, LSTM has feedback connections, it can not only process single data points, but also entire sequence of data, time series forecasting nicely fits into one or multi class problem.

Only simple LSTM layer was able to implement as first phase, neural network output is feeding back at a time frequency t to the input of the same network layers at a time t+1 look 

Three layers of LSTM with dropout has been designed using Keras , for each layer weight(W) has been gradually increased from 50 to 400,

LSTM Architecture for sales prediction problem



**Data preprocessing for LSTM**

Select only 28 days prediction to start with, normalize to prepare time series data the goal of normalization is to change the value of numeric columns in the dataset to use a common scale , without distorting differences in the time range of values, for sales data ranges could.

vary widely, WalMart data has been normalized using minmax scale of –1 and 1,Plot the differences to make sure only scale has changed not data distorted.

Create time series data sequence into input features and labels with techniques to adpot time series, RNN requires data into windows of sequences and lables, window sequence of 28 days created so that data looks like a series of data.

Training and test data were split into 70 to 30 ratio training data was fed into RNN LSTM layer.

#### **LSTM parameters:**

optimizer = 'adam',

loss = 'mean\_squared\_error'

epoch\_no=32

batch\_size\_RNN=44

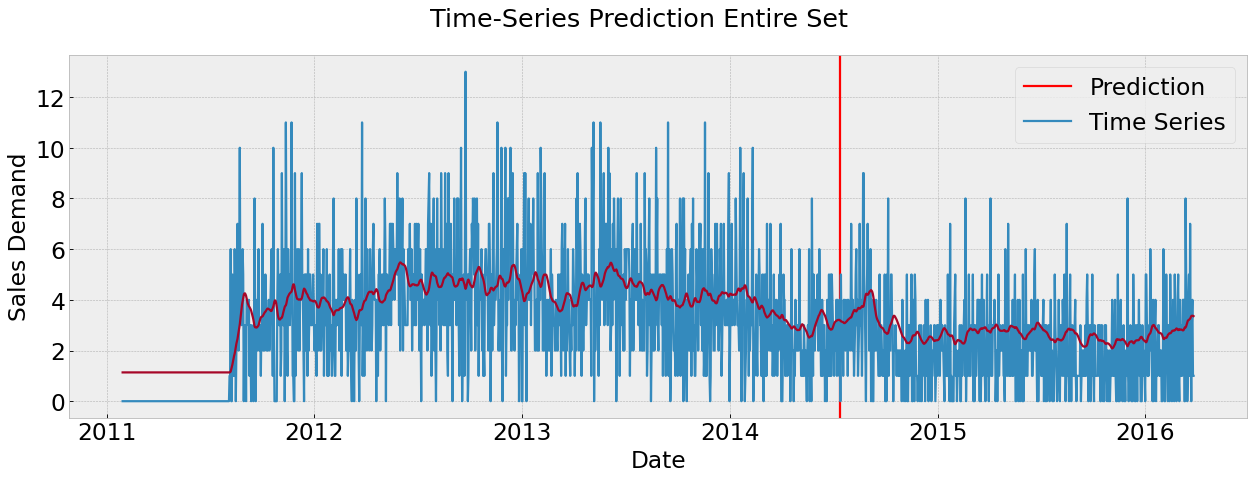
#### **LSTM metrics**

Accuracy:100

RSME: 2.0252094

### **Prediction**

Time series prediction of entire dataset for 28 days prediction as shown in the code, prediction for an item wise, item code has a state, store and department and itemid , X axis as entire time frame and Y axis sales demand



Multiclass and feature engineering.

1. Predicting Uncertainty.
2. Introduction

When making a prediction about the data in the real world, it is good to estimate how sure is the model of the prediction that it made. This is true of the model that have direct impact on the lives of the people. Fields like autonomous driving cars, stock market, health care, construction etc. have a direct impact on the lives of people. These field have seen many advancements and many machine learning; AI techniques are employed in these areas.

1. Sources of Uncertainty

When dealing with uncertainty we are primarily concerned with two types uncertainty.

* Aleatoric Uncertainty – This kind of uncertainty is present in the process and is an integral part of it. For example if we rebound a rubber ball from the ground, it is not 100% sure that ball would rebound to the same height every time. The factors like air drag, point of contact could add variance to the rebound parameter and hence the uncertainty in the prediction.
* Epistemic Uncertainty – This uncertainty is about the prediction of the model and could be attributed to the data sampling, feature selection, model parameter tuning. This could be minimized with appropriate analysis of the data and model parameters. For example selecting the important feature could lead to better prediction value and lesser variance. We have tried to focus our study on this type only.

1. Data Collection

The data for the uncertainty prediction is essentially the same as used for sales forecasting. Since, we are calculating the uncertainty for the forecasting, it quintessential we use the same data to predict the uncertainty to keep both the models in sync with their results. Further the algorithm used is also the same for both the model, a 3-layer LSTM deep neural network.

The data model: We are running a neural network on a system with limited compute power, we choose to forecast a single item form a department within one of the stores, HOUSEHOLD\_1\_122\_CA\_3. The product is a household category item, product category 1\_122 for the California store 3 of Walmart. The data is a normalized in the range -1 to 1 and indexed by date.

HOUSEHOLD\_1\_122\_CA\_3

2011-01-29 0

2011-01-30 0

2011-01-31 0

2011-02-01 0

2011-02-02 0

Historical\_data = 1913

Train\_data = 0.7\*1913

Test\_data = 0.3\*1913

The historical sales data is provided for the last 1913 days, we tried to split the data in 70:30 ratio. 70 % of the data used for training and 30 % used for testing.

1. Machine learning attempt
2. Maximum Likelihood Method (MLE) interval estimate

We assume the mean of forecasted sales is a gaussian distribution i.e. and is normally distributed. We create two models:

* Mean model: m­µ This estimates the mean response ­i.e. the mean class conditional density. The train data is split equally 0.5\*Train\_data each for mean and variance model.

The mean model is trained using the first half of the training data and the following equation

A picture containing object, clock

Description automatically generated

* Variance model: m This model estimates the variance i.e. .

The mean model m­µ is used to make prediction on the second half of the data to calculate the squared residual as the dependent variable.

Variance model equation

A picture containing object, clock

Description automatically generated

The uncertainty using MLE method is calculated

A close up of a logo

Description automatically generated

Where **α** is the confidence interval of prediction. For confidence interval 95 % we plot the gaussian mean variance uncertainty Fig 1.

A screenshot of a cell phone

Description automatically generated

Fig 1

We notice from the plot that the highly inaccurate predictions have higher intervals. Which goes to say that the inaccurate predictions have difficulty in predicting the range of confidence interval.

1. Quantile Range interval

We use the quantiles to predict the range of the uncertainty. A lower quantile and a higher quantile is used as confidence interval. We train two different regression model lower quantile model with 0.025 percentile and a higher percentile with 0.9725 percentile. A 95% prediction interval.

The quantile regression loss function

A close up of a mans face

Description automatically generated

[*Quantile Regression Loss Function*](https://arxiv.org/pdf/1806.11222.pdf)

Plotting the quantile range uncertainty estimation for the 28 days for the item category household, product category 122 and store 3 California.

A close up of a mans face

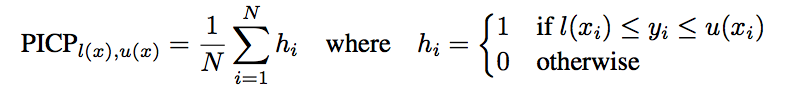
Description automatically generated

Notice the prediction that do not match well with the actual data have wider prediction intervals. This shows that the feature values have low information to produce the information.

The interval estimates are sensitive to the training sample of the mean and variance in case of gaussian estimation. And, the interval estimates in similar way are sensitive to the lower and upper bound quantiles which are also bound to the samples size and the droupout factor.

We define two metric that helps us capture the quality of our intervals.

* Prediction Interval Coverage Probability (PICP) - this tells us the percentage of time an interval contains the actual value of the prediction.



* Mean Prediction Interval Width – It is the average width of a predicted interval.

A close up of a clock

Description automatically generated

The plot of MPIW, PICP plot for range quantile and mean variance

Our goal is to maximize PICP, while minimizing MPIW in order to get high quality intervals.

A screenshot of a cell phone

Description automatically generated

Conclusion and Future Work

Reference

<https://arxiv.org/pdf/1806.11222.pdf>

<https://arxiv.org/abs/1806.11222>

<https://medium.com/comet-ml/estimating-uncertainty-in-machine-learning-models-part-3-796efba4a209>

<https://www.kaggle.com/c/m5-forecasting-accuracy>

<https://www.kaggle.com/c/m5-forecasting-uncertainty>

<https://www.kaggle.com/omershect/learning-pytorch-lstm-deep-learning-with-m5-data>

<https://www.kaggle.com/omershect/learning-pytorch-lstm-deep-learning-with-m5-data>