Konstantino Sparakis : U47131572
Shuwen Sun: U74918696
November 4, 2016
CS591

Mid Progress Report 2

GITHUB: https://github.com/AWS-Spot-Analysis/spot-analysis

Attached to the email are:
Sample data from api,
Sample data from archives,
And our working code in jupyter notebook

Abstract

We are analysing the price marketplace of AWS EC2 spot instances. Through our research, we plan to study the marketplace which consists of bidding for virtual resources. Our goal, is to discover patterns and test hypothesis using the data provided from this market place. We hope to be able to solve question like is there the prices in different marketplaces/locations highly correlated or not correlated, are we able to predict the volatility of the bid market for certain machines given. This kind of research will benefit the industry community as well as the academia. Some ongoing research claim that computer scientists should take a look into the bidding strategy, meanwhile others suggest working on virtual resources checkpointing and migrations. This project will provide some fresh view on how to deal these spot marketplace and the trending topics in cloud. Also, we hope that our analysis will help people understand the marketplace better allowing people to save money, and give people a starting point regarding future research.

Introduction

Our dataset consists of multiple time series of bid pricing for Amazon clouds EC2 Spot instances. EC2 is a service that allows you to purchase and use a custom virtual machine on an hourly basis that runs in a remote server and location, the cloud. EC2 virtual machines consist of different specs, allowing users to choose everything from RAM, vCPU's, ssd memory, OS and more options. Amazon also has multiple data centers throughout the world and these are considered regions.

What amazon introduced with Spot instances is that since not all their computing resources are being used up at any given moment. You can bid on these leftover machine resources starting at about 1/10th the original price. As the resources available is variable with decrease in supply, the demand goes up and so does the price of the resources. If you get outbid your instance is shutdown after 2 minutes. The way the bidding works is a follows, I choose the maximum amount I want to pay per hour. Depending on how many resources are available and what people are willing to pay the market fills up, by servicing the people who are willing to pay more first then to the people who want to pay less. As this fills up if the resources are running low then the price moves up because they need to cut out the people who are not willing to pay more.

Hypothesis

Now with this data is where it gets tricky and creative. We have decided to only look at machines whose operating system is Linux, this is due to more data being available and to

tighten our scope. We have to in a sense massage the data in order to start seeing what techniques return interesting results worth reporting.

For this project, the problem we are specifically interested in is the correlation of virtual machines across different regions. Amazon separate the region into us-west-1a, us-east-1a, etc. **The hypothesis we have in mind is that those market prices in different region are highly uncorrelated, but inside each marketplace, they are correlated.**

Our goal with our analysis and answering these hypothesis is to help inform people on the environment of the market. We hope to uncover things that will allow people to choose a machine, in an area with a better idea of how likely they are to get outbidded, and what days might be the better days for them to use spot instances to get a cheaper pricing, or is it in fact even worth just consistently paying for spot instance and move away from original instances?

Dataset Collection and Explanation

What our approach has consisted of is first off hunting for data, aws only provides 90 days of most recent data, archives exist but finding ones that were still serving their data was difficult but we managed to pull through. We obtained data ranging from 2014-04-17 to 2015-08-24 that looks like such:

Title	Price	Timestamp	Machine type	OS	AWS Region
SPOT INSTANCE PRICE	0.107100	2014-03-27T 07:33:04-070 0	c3.2xlarge	Linux/UNIX	us-west-1a

The price is the new price set by the market, at that timestamp. Since Aws has servers in different areas of the world we also get the AWS region data to classify where the machine resources are located that we are bidding for. Since different users need different amounts of resources there exists machine type. For example one machine will have more cpus, while another has more ram. These are pre-defined setups that are described on the aws website. We can also choose what operating system we want and this is why the OS data exists. But we will only be working with Linux so we can cut this data piece out.

We also have used the aws terminal client and api to grab the previous 90 days. Which a sample of this is included in our report. It comes in JSON format so some more processing needs to be done but this is rather simple.

Using this command:

```
aws ec2 describe-spot-price-history --instance-types m1.xlarge --start-time
2014-01-06T07:08:09 --end-time 2014-01-06T08:09:10
```

Retrieving:

...
{
"Timestamp": "2016-11-06T08:01:04.000Z",
"ProductDescription": "SUSE Linux (Amazon VPC)",
"InstanceType": "m1.xlarge",
"SpotPrice": "0.133500",
"AvailabilityZone": "us-east-1d"
}
...

What each data point shows is a price change in the market and at what time this occurred. The problem with this data is that price changes happen differently so in order to do time series analysis we need to first resample the data, we have chosen to resample by an hourly basis, to normalize all the data to one fit.

Techniques used in this project

Time Series Correlations

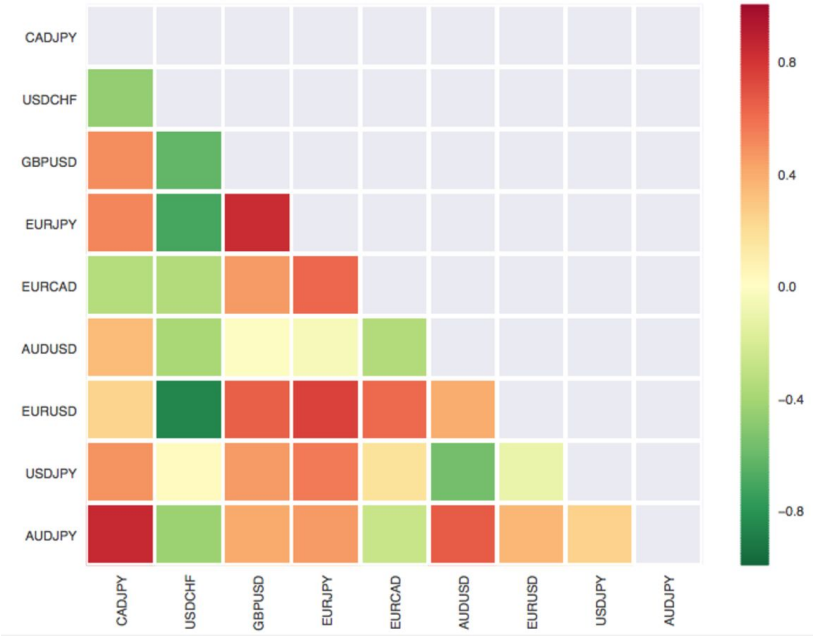
We have decided that one of the biggest tells in this project would be looking at the correlations of the different markets. What we want to find are the following using this technique:

- How similar is the market for the same machine in a different location?
- How similar are different machines within the same market?

By finding correlations we hope to uncover patterns that can be used to advantage a user. For example, if the same machine in different locations has an unrelated market then a user can choose to move their machine to the different location if they get outbidded and still use the resources at the price they most hope for.

Heat map

We can use a heat map to help us better visualize what the correlations look like.



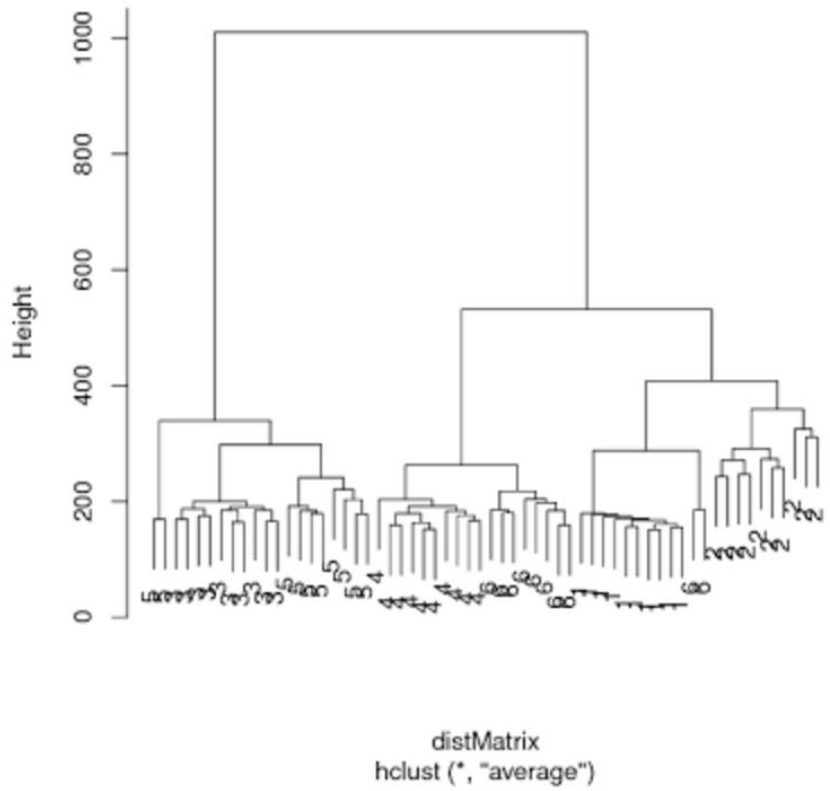
What this heat map for example shows is the redder the box the more correlated the times are, the greener the item it means it has a negative correlation which means that when item goes up the other goes down.

Time Series Clustering

We will also be attempting to use techniques of time series clustering to compare our results with correlations. By this we mean partitioning time series data into groups based on similarity or distance, so that time series in the same cluster are similar. This will show us what regions are similar, what machines are similar and so on. Here we can use different clustering algorithms such as K-means, Hierarchical and so on. We will also attempt to use Dynamic Time Warping (DTW) to find optimal alignment between two time series.

For this method we do not have to normalize the times, which shows us details about how often the price changes in that time series and is something lost with the time normalization done in the correlation code.

Using this we can obtain a tree like the image displayed below to show us the similarities between nodes. Where each node is a time series.



Progress Since Report 1

We have run into a few issues since report 1. We have decided trying to predict what the future market is too difficult of a task, since even stock market prediction is truly unreliable. We have decided instead to focus on finding interesting correlations and using different techniques to do so.

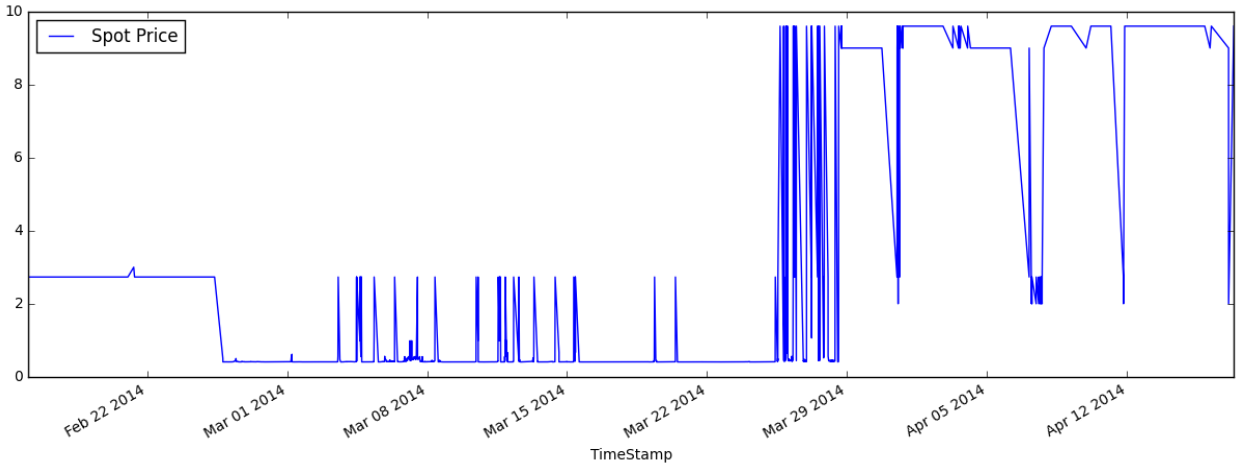
We also have the issue where our data is too big. Unzipping our data comes out to over 70gb of data and we don't have enough memory on our computers to handle the file size as we are working off of laptops. Also the processing time for such huge files is too long. So what we have decided to do is start out by taking a small sample and get our tools to find correlations. After this we will concentrate on processing as much of this data as we can.

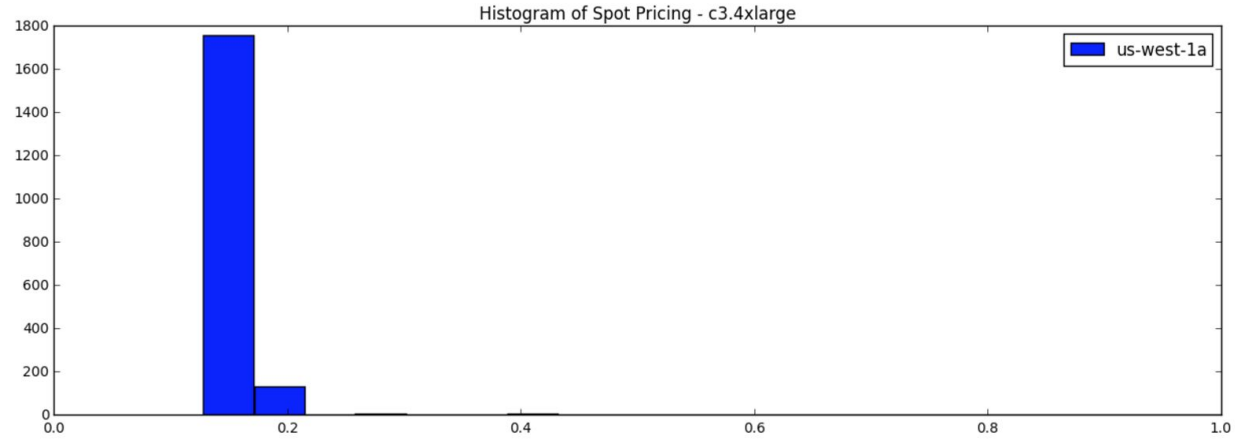
What we have hit as a wall and have been working hard at debugging is that in order to do these time series analysis we need to normalize the time data, what I mean by that is that the data is only recorded when the price change happens. Because of this the time series do not have the same times, and we have been working on code that resamples the times by hour. But we run into issues where 1 markets time series has missing samples compared to anothers.

Because of this we can't use the correlation function. We have been working on a solution to clear out all missing values so that the time series line up perfectly. Once we have this working we can get correlations, and will move onto clustering time series.

We have already created code to get the raw data into data frames, and here are some graphs to prove this.

Here are some example graphs:





Results and Discussion

To be completed with the final report.

Conclusion

To be completed with the final report

Below is our in-progress work:
AWS-Spot-Analysis/spot-analysis