Business Intelligence
& Data Warehousing
MSIS 2621 & OMIS 3386

Jonathan Wu

Team Kickass

Bhakti Mohadkar
Frederick Su
Mike Greco
Sagar Botta
Sangramsingh Kardekar

Contents

Executive Summary:

Team Kickass designed and developed a business intelligence solution that aims to shed light on the specific factors that influence housing prices in the San Francisco Bay Area. We considered regional and specifically corporate factors that directly contribute towards the highly volatile real estate in the area. We analyzed data from 2010-2015, aiming for timely and actionable results for the current market. The data analyzed includes weather data, unemployment, a variety of stock prices, and the housing price index. We have identified which factors are most statistically correlated to the change in average housing prices across each major locale within the San Francisco Bay Area.

Collecting and relating this data has been a challenging process. Each data source had it's own quirks and inconsistencies, and transforming each one into a usable format took a considerable amount of effort. Housing prices posed a particular challenge, with many of the best data sources remaining locked away behind paywalls. Fortunately the Federal Housing Finance Agency publishes a Housing Price Index which gave us precisely what we needed.

Our final deliverable is a series of visualisations for each locale mapping the average percent difference for each of our measures over a configurable time span. We have calculated the percentage change for each of our measures so that we may directly compare elements on a common scale and axis. This has allowed us to use a variety of statistical methods to compare data trends including: linear regression, the comparison of regression slopes, and the analysis of covariance between two regressions.

Running an analysis of variance on each linear regression of change allows us to determine how closely correlated each data set is to housing price in an objective fashion. Comparing the slope of each regression allows us to determine the scaling factor of each related element and whether the element is positively or negatively correlated to the housing price.

This analysis has provided us with 3 key insights:

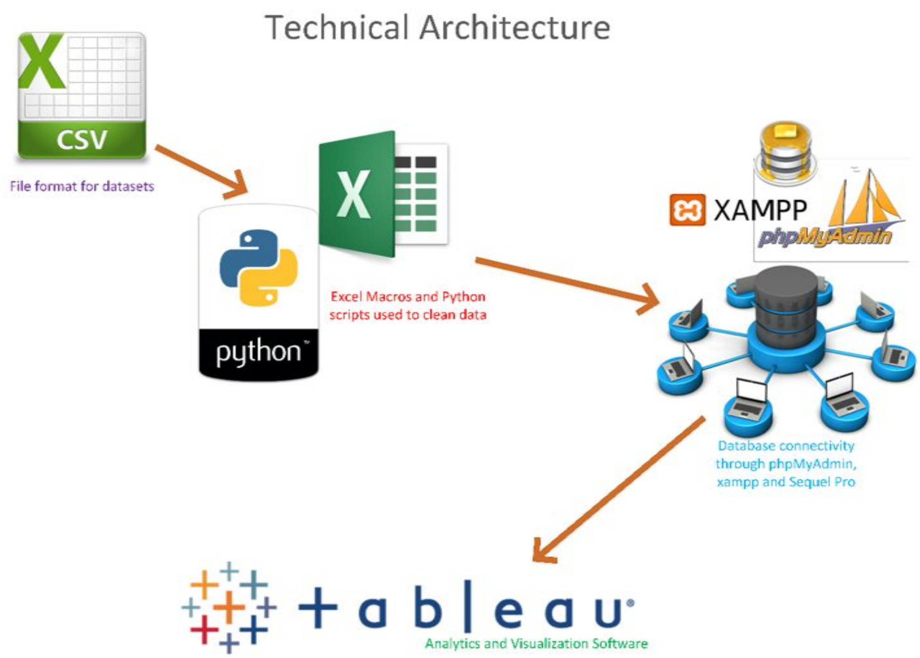
1) The housing price index is very highly correlated with the stock performance of Yahoo! and HP Inc. across the entirety of the San Francisco Bay Area.
2) Although the average housing price is variable by Bay Area locale, the rate of change for each area is very similar.
3) The housing price index is most significantly affected by the stock prices of large institutions, startups have little to no impact on the average price of real estate.

List of Team Members and Responsibilities:

Team Member	Responsibilities
Mike Greco	*Project Manager, Technical Architect, and Implementer.* Bringing together all aspects of the project. Filling in the gaps and verifying correctness at each stage. Resident do-it-all.
Fred Su	*Business Analyst, Data Sourcer.* Identify usable data sources, converting high-level requirements into quantifiable ones. Ensuring the implementation matches the concept.
Bhakti Mohadkar	*Project Coordinator, Data integrator,Technical Writer.* Worked on gathering data sources and data cleansing. Maintained data guidelines and data dictionary. Drove project completion and led completion of project deliverables. Documented working process for creation of project report.
Sangramsingh Kardekar	*Technical Architect, Data Sourcer.* Identified viable technologies by conducting trial runs before actual implementation. Researched and summarized product features used in the implementation of deliverable.
Sagar Botta	*Developer, Data Sourcer.* Automating data cleansing and integration. Manipulating data sources and deriving insights from visuals. Code guy.

Technical Architecture:

Our technology stack includes: Python for data conversion, MySQL for the database, and Tableau Desktop for analytics. Additional tooling includes: Microsoft Excel for CSV manipulation, phpMyAdmin and Sequel Pro for database access, and vi for text manipulation.



Challenges and issues with your project and group:

Our project had quite a few significant challenges along the way. Our most significant challenge was in the procurement of free data sets that would satisfy our needs. Many desirable data sources tangential to the real estate industry require expensive subscriptions to access. Other datasets could not be converted to a usable format in any reasonable amount of time, forcing our team to scrap them.

Our team also suffered from a lack of development experience. As a result, team members capable of large scale data transformation became critical path. Similar challenges were faced in the creation of our SQL queries and the SQL based percentage change calculations. All delays in project timeline could be attributed to the bottleneck of development experience.

Cumulatively the team contributed approximately 140 person-hours of work toward the project. 10 hours of work can be attributed to defining the project direction and staging of the work environment. 20 hours of work can be attributed to gathering appropriate data sources. Cleansing and transformation of the data required 60 hours of work, including the development of data transformation scripts. Integrating data took 20 hours of work, and the creation of documentation and reporting materials took a final 30 hours of work.

Changes from original project proposal:

Our original project proposal called for data to be culled from:
Google trends, Yahoo finance, Zillow or Trulia, The DOT, Weather Underground, the Department of Labor, The FBI, the US Census,The California Department of Education and WalkScore.com.

After deeper investigation we discovered that the APIs for Zillow, Trulia, and WalkScore.com required specific locations to be provided to gather any results. This would require we compile a list of all street addresses within our target locations to gather results for these data sources. This was a larger technical hurdle than we were equipped to deal with, and as a result we found an alternative data source in the Housing Price Index provided by the Federal Housing Finance Agency.

Traffic data from the Department of Transportation proved to be more difficult to parse than our collective skillset would accommodate. Traffic volume statistics were reported “before” and “after” selected intersections on highways, with the next level of location granularity being county. The level of complexity in transforming this data was deemed too great to accomplish with the technical resources we had available and was therefore dropped.

The source of weather data was initially Weather Underground. Further investigation into the Weather Underground API revealed that historical data required an expensive subscription. Weather data was instead sourced from the United States Historical Climatology Network, provided free of charge by the US Department of Energy.

We initially intended on incorporating school rating data into the model, and planned on creating a mapping between district name and ZIP code. Upon deeper investigation we discovered that district name was not consistently comparable to city name. Manually mapping this data proved to be too labor intensive to sustain and the data was dropped as a result.

The last difference from project proposal came in the form of crime data. Crime data was available at a minimum resolution of one year. This low resolution introduced scaling issues which could not be resolved within the time frame allotted, causing us to drop crime as a datapoint.

Data Transformation and Loading:

All of the data leveraged in our project required cleansing and transformation to be actionable. Date was our most commonly used primary and foreign key. The preferred date format for MySQL is YYYY-MM-DD. Unfortunately Microsoft Excel does not cooperate with this date format easily. Instead of using a custom format in Excel, the team chose to automate the process by using a python conversion script to convert the format as the last step prior to loading the data.

We also had data resolution issues, specifically with the Housing Price Index and stock data. HPI was available per-quarter only due to the relatively slow rate usually observed in an index of this fashion. To better analyze the trends in this data over time, we extrapolated HPI to representative daily values using a technique called linear interpolation. By filling in the missing values we were able to more efficiently map and plot HPI data to rapidly changing measures such as stock price and weather data.

Location posed a particular issue, as none of our location based data was reported in a consistent format. The housing price index is reported by CBSA, or Core Based Statistical Area. Most other location based data is reported by ZIP code, which is not directly compatible with CBSA. Since CBSA is the larger area, location data based on ZIP codes required aggregation and mapping to be comparable.

Our last major data transformation hurdle was scaling the data to comparable figures. The HPI index is given as a number without a specified unit. Stock prices are generally USD. Weather information comes in the form of degrees and inches. One unifying measure out of all these incompatible data types is percentage change. Calculating the percent change of each figure allows us to directly compare each rate of change on the same axis.

The percentage change for each time variant element was calculated with the following SQL query while loading the fact table:

```
LEFT JOIN (Select S1.Date as "Date", S1.place_id, ((S1.hpi - S2.hpi) /
S2.hpi)*100 as "Change" FROM HPI as S1 INNER JOIN HPI as S2 ON S1.Date =
(ADDDATE(S2.Date, INTERVAL 1 DAY)) and S1.place_id = S2.place_id and
S1.place_id = 34900) AS CHPI on CHPI.Date = WEATHER.Date
```

Tableau provided some utilities to perform SQL joins and calculations on our behalf, but the performance of these features was not acceptable to rapidly iterate on the data set. As a result, we crafted SQL statements to calculate this data and load it prior to any analysis or manipulation in Tableau.

SQL Statements:

The primary SQL statement used was in the creation of the fact table:

```
CREATE TABLE FACT_34900 AS (
SELECT WEATHER.*, CFB.Change As 'FB % Change', CGOOGL.Change As 'GOOGL % Change', CHP.Change
As 'HP % Change', CORCL.Change As 'ORCL % Change', CYHOO.Change As 'YHOO % Change',
CAAPL.Change as 'AAPL % Change', CHPI.Change as 'HPI % Change', UNEMPLOYMENT.UNEMPLOYMENT
FROM WEATHER
    JOIN UNEMPLOYMENT
       ON UNEMPLOYMENT.CBSA = 34900
       LEFT JOIN (Select S1.Date as "Date", S1.place_id, ((S1.hpi - S2.hpi) / S2.hpi)*100 as
"Change" FROM HPI as S1 INNER JOIN HPI as S2 ON S1.Date = (ADDDATE(S2.Date, INTERVAL 1 DAY))
and S1.place_id = S2.place_id and S1.place_id = 34900) AS CHPI
              on CHPI.Date = WEATHER.Date
       LEFT JOIN (Select S1.Date as "Date", ((S1.Price - S2.Price) / S2.Price)*100 as "Change"
FROM HP as S1 INNER JOIN HP as S2 ON S1.Date = (ADDDATE(S2.Date, INTERVAL 1 DAY))) AS CHP on
CHP.Date = WEATHER.Date
       LEFT JOIN (Select S1.Date as "Date", ((S1.Price - S2.Price) / S2.Price)*100 as "Change"
FROM ORCL as S1 INNER JOIN ORCL as S2 ON S1.Date = (ADDDATE(S2.Date, INTERVAL 1 DAY))) AS
CORCL on CORCL.Date = WEATHER.Date
       LEFT JOIN (Select S1.Date as "Date", ((S1.Price - S2.Price) / S2.Price)*100 as "Change"
FROM YHOO as S1 INNER JOIN YHOO as S2 ON S1.Date = (ADDDATE(S2.Date, INTERVAL 1 DAY))) AS
CYHOO on CYHOO.Date = WEATHER.Date
       LEFT JOIN (Select S1.Date as "Date", ((S1.Price - S2.Price) / S2.Price)*100 as "Change"
FROM FB as S1 INNER JOIN FB as S2 ON S1.Date = (ADDDATE(S2.Date, INTERVAL 1 DAY))) AS CFB on
CFB.Date = WEATHER.Date
       LEFT JOIN (Select S1.Date as "Date", ((S1.Price - S2.Price) / S2.Price)*100 as "Change"
FROM AAPL as S1 INNER JOIN AAPL as S2 ON S1.Date = (ADDDATE(S2.Date, INTERVAL 1 DAY))) AS
CAAPL on CAAPL.Date = WEATHER.Date
       LEFT JOIN (Select S1.Date as "Date", ((S1.Price - S2.Price) / S2.Price)*100 as "Change"
FROM GOOGL as S1 INNER JOIN GOOGL as S2 ON S1.Date = (ADDDATE(S2.Date, INTERVAL 1 DAY))) AS
CGOOGL on CGOOGL.Date = WEATHER.Date
WHERE WEATHER.Date > "2010-01-01"
);
```

An additional SQL statement was used to create aggregate dimension tables for plotting:

```
CREATE TABLE DIMENSION_34900 AS (
SELECT WEATHER.*, FB.Price As 'FB', GOOGL.Price As 'GOOGL', HP.Price As 'HP', ORCL.Price As
'ORCL', YHOO.Price As 'YHOO', AAPL.Price as 'AAPL', HPI.hpi, UNEMPLOYMENT.UNEMPLOYMENT
FROM WEATHER
    LEFT JOIN FB
        ON FB.Date = WEATHER.Date
    LEFT JOIN GOOGL
        ON GOOGL.Date = WEATHER.Date
    LEFT JOIN HP
        ON HP.Date = WEATHER.Date
    LEFT JOIN ORCL
        ON ORCL.Date = WEATHER.Date
    LEFT JOIN YHOO
        ON YHOO.Date = WEATHER.Date
    LEFT JOIN AAPL
        ON AAPL.Date = WEATHER.Date
    LEFT JOIN HPI
        ON HPI.Date = WEATHER.Date and HPI.place_id = 34900
    LEFT JOIN UNEMPLOYMENT
       ON UNEMPLOYMENT.CBSA = 34900
WHERE WEATHER.Date > "2010-01-01");
```

Data Validation
After data was loaded into the environment, we validated the data by verifying the checksum. We performed a `select sum(column) from table` function in SQL and compared this against the sum generated in Microsoft Excel to validate data. Fact table data was spot checked against manual calculations and was individually assessed for completeness.

Method of Analysis:
We used a number of statistical analysis methods to make accurate determinations from the dataset that we have compiled. The baseline tool of analysis has been linear regression. Linear regression has allowed us to plot the general trend of a measure's change. Calculating the formula of linear regression has also allowed us to perform an analysis of variance or ANOVA test, giving us an objective measure to assess the significance of an individual measure in our model. Finally the slope of the formula of linear regression for a measure allows us to compare the impact of variance in a measure to the model as a whole. Tableau made generating the linear regression for each measure straightforward, using the trend line feature. Once trend lines have been generated, Tableau also allowed us to export the raw statistical data for deeper analysis.

Results:
Bay Area housing prices are highly volatile, but largely homogenous. Over the time period assessed, from 2010-2015, when the price of housing goes up in one locale, it will go up in the entire San Francisco Bay Area. As a result, no measures we incorporated affect one locale more strongly than another.

The largest influencers in our analysis on Bay Area housing prices are the stocks of HP and Yahoo. There is a strong positive correlation between these stocks and the housing market in the area, so a dramatic rise or drop in either stock will act as a predictor for housing value.

Facebook, a large company with a relatively recent IPO, contributed in a small but measurable way to the general upward trend in HPI. In contrast small startup companies with a recent IPO had no statistical correlation with HPI.

HPI Source File:
http://www.fhfa.gov/DataTools/Downloads/Documents/HPI/HPI_PO_state.xls

hpi_type	hpi_flavor	frequency	level	place_nan	place_id	yr	period	index_nsa	index_sa
traditional	all-transactions	quarterly	MSA	Napa, CA	34900	1978	4	34.63	
traditional	all-transactions	quarterly	MSA	Napa, CA	34900	1979	1	35.77	
traditional	all-transactions	quarterly	MSA	Napa, CA	34900	1979	2	37	
traditional	all-transactions	quarterly	MSA	Napa, CA	34900	1979	3	39.57	
traditional	all-transactions	quarterly	MSA	Napa, CA	34900	1979	4	39.99	
traditional	all-transactions	quarterly	MSA	Napa, CA	34900	1980	1	43.25	
traditional	all-transactions	quarterly	MSA	Napa, CA	34900	1980	2	43.06	
traditional	all-transactions	quarterly	MSA	Napa, CA	34900	1980	3	44.71	
traditional	all-transactions	quarterly	MSA	Napa, CA	34900	1980	4	45.77	

HPI Data Cleansing Process:

The most consistent data was based off of "traditional" and "all transactions", so all other types were discarded. The seasonally adjusted index (index_sa) was incomplete for some date ranges, therefore we used the non-seasonally adjusted index (index_nsa).
The data file was then split by location for easier processing.

year	period	place_id	place_nan	index_nsa
2001	1	34900	Napa	160.98
2001	2	34900	Napa	166.96
2001	3	34900	Napa	171.68
2001	4	34900	Napa	172.75
2002	1	34900	Napa	179.28
2002	2	34900	Napa	184.58
2002	3	34900	Napa	191.37
2002	4	34900	Napa	196.99
2003	1	34900	Napa	203.46

Converting the quarterly data into individual days was done via a pair of python scripts. The first script, getmonths.py, converted quarters into months, leaving the HPI field blank for months without associated data. The second, getdays.py, fills in day-level data for each given month. This script is calendar-accurate, accounting for leap year and other date variations.

With our template for each place_id in place, we used a technique called Interpolation to fill in the empty data points. Instead of coding this manually, we used the following tool to do this:
http://www.digdb.com/excel_add_ins/fill_blank_cells_interpolate/

Date	place_id	place_name	hpi
1/1/2001	34900	Napa	160.98
1/2/2001	34900	Napa	161.05
1/3/2001	34900	Napa	161.11
1/4/2001	34900	Napa	161.18
1/5/2001	34900	Napa	161.25
1/6/2001	34900	Napa	161.31
1/7/2001	34900	Napa	161.38
1/8/2001	34900	Napa	161.45
1/9/2001	34900	Napa	161.51

Stock Source Files:
http://finance.yahoo.com/q/hp?s=AAPL+Historical+Price

Date	Open	High	Low	Close	Volume	Adj Close
12/3/2015	116.55	116.79	114.22	115.2	41476500	115.2
12/2/2015	117.34	118.11	116.08	116.28	33199000	116.28
12/1/2015	118.75	118.81	116.86	117.34	34701000	117.34
11/30/2015	117.99	119.41	117.75	118.3	37658700	118.3
11/27/2015	118.29	118.41	117.6	117.81	13023700	117.81
11/25/2015	119.21	119.23	117.92	118.03	21388300	118.03
11/24/2015	117.33	119.35	117.12	118.88	42803200	118.88
11/23/2015	119.27	119.73	117.34	117.75	32482500	117.75
11/20/2015	119.2	119.92	118.85	119.3	34287100	119.3

We used only Adj Close and Date from the stock data. Adjusted Close accounts for any inconsistency in stock data such as a stock split.

All stock data followed a consistent format after cleansing:

Date	Adj Close
10/28/2015	119.27
10/27/2015	114.55
10/26/2015	115.28
10/23/2015	119.08
10/22/2015	115.5
10/21/2015	113.76
10/20/2015	113.77
10/19/2015	111.73
10/16/2015	111.04

Weather Source File:

Historical weather details were provided by the United States Historical Climatology Network.
http://cdiac.ornl.gov/epubs/ndp/ushcn/ushcn_map_interface.html
http://cdiac.ornl.gov/cgi-bin/broker?_PROGRAM=prog.climsite_daily.sas&_SERVICE=default&id=040693&_DEBUG=0

Date	PRCP (in)	TAVE (F)	TMAX (F)	TMIN (F)
1/1/2000	0	48	50	45
1/2/2000	0	47	53	40
1/3/2000	0	49	55	42
1/4/2000	0.05	49	55	42
1/5/2000	0	48	56	40
1/6/2000	0	50	59	41
1/7/2000	0	50	58	41
1/8/2000	0	51	56	45
1/9/2000	0	51	56	46
1/10/2000	0.02	52	57	47

The Climatology Network provides a number of options to generate a file including date range, selected columns, and even file name. No additional data cleansing was required.

Unemployment Source File:
http://zipatlas.com/us/ca/city-comparison/unemployment-rate.htm

#	Zip Code	Location	City	Population	% Unemployment	National Rank
1	95232	38.358464	Glencoe, (	17	100.00%	#5
2	96119	41.042003	Madeline,	70	66.66%	#21
3	90822	33.778436	Long Beac	422	66.27%	#23
4	95424	38.970739	Clearlake	91	64.28%	#26
5	96108	41.750354	Davis Cree	91	55.17%	#48
6	95387	37.546007	Westley, (	897	53.64%	#49
7	95655	38.549822	Mather, C	914	52.91%	#53
8	90013	34.044639	Los Angel	9,727	50.19%	#60
9	93447	35.666506	Paso Robl	794	50.00%	#61
10	95981	39.584580	Strawberr	98	47.45%	#76

Cleaning this data consisted of removing: #, Location, City, Population and National Rank. After doing so, we generated averages for all ZIP codes within a CBSA.

42100	0.062941
41884	0.046182
42034	0.024286
34900	0.041818
41940	0.047833
36084	0.053038
44700	0.108824

ZIP Code Source File:
https://www.census.gov/econ/cbp/download

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	zip		name	empflag	emp_	emp	qp1_nf	qp1	ap_nf	ap	est	city	stabbr	cty_name
2	94102	94102	SAN FRANCISCO, CA		G	30071	G	320065	G	1376148	1702	SAN FRANCISC	CA	SAN FRANCISCO
3	94103	94103	SAN FRANCISCO, CA		G	56580	G	954911	G	4090074	2770	SAN FRANCISC	CA	SAN FRANCISCO
4	94104	94104	SAN FRANCISCO, CA		G	39581	G	1394231	G	5485450	2085	SAN FRANCISC	CA	SAN FRANCISCO
5	94105	94105	SAN FRANCISCO, CA		G	89115	G	3781626	G	12279293	2217	SAN FRANCISC	CA	SAN FRANCISCO
6	94107	94107	SAN FRANCISCO, CA		G	49119	G	958007	G	4351341	2349	SAN FRANCISC	CA	SAN FRANCISCO
7	94108	94108	SAN FRANCISCO, CA		G	26605	G	502297	G	1917570	1750	SAN FRANCISC	CA	SAN FRANCISCO
8	94109	94109	SAN FRANCISCO, CA		G	25582	G	277375	G	1212354	1724	SAN FRANCISC	CA	SAN FRANCISCO
9	94110	94110	SAN FRANCISCO, CA		G	22125	G	240702	G	1079104	1939	SAN FRANCISC	CA	SAN FRANCISCO
10	94111	94111	SAN FRANCISCO, CA		G	51450	G	1894435	G	6732155	2525	SAN FRANCISC	CA	SAN FRANCISCO
11	94112	94112	SAN FRANCISCO, CA		G	5492	G	47430	G	189806	752	SAN FRANCISC	CA	SAN FRANCISCO
12	94114	94114	SAN FRANCISCO, CA		G	7710	G	77379	G	323207	995	SAN FRANCISC	CA	SAN FRANCISCO
13	94115	94115	SAN FRANCISCO, CA		G	17498	G	275075	G	1095737	1190	SAN FRANCISC	CA	SAN FRANCISCO
14	94116	94116	SAN FRANCISCO, CA		H	5401	H	63754	H	256816	644	SAN FRANCISC	CA	SAN FRANCISCO
15	94117	94117	SAN FRANCISCO, CA		G	12296	G	120066	G	533062	879	SAN FRANCISC	CA	SAN FRANCISCO
16	94118	94118	SAN FRANCISCO, CA		G	12469	G	145696	G	600437	1237	SAN FRANCISC	CA	SAN FRANCISCO

To cleanse ZIP code data, we removed columns: name, empflag, emp_f, emp, qp1_nf, qp1, ap_nf, ap, est, and city. We then mapped ZIP Code to CBSA using a VLOOKUP in excel.

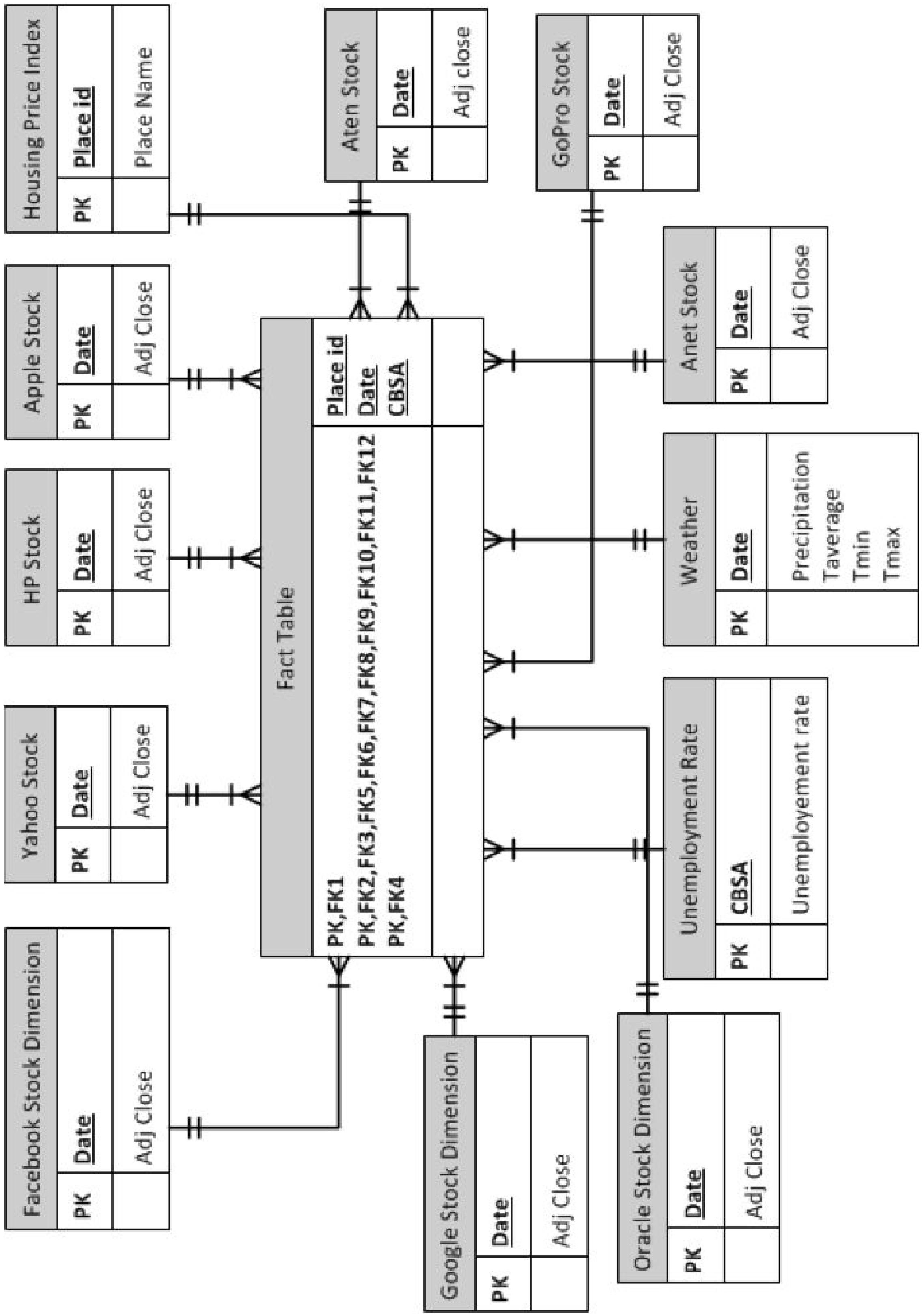
zip	CBSA	cty_name
93925	41940	MONTEREY
94002	41884	SAN MATEO
94005	41884	SAN MATEO
94010	41884	SAN MATEO
94011	41884	SAN MATEO
94014	41884	SAN MATEO
94015	41884	SAN MATEO
94017	41884	SAN MATEO

CBSA Source File:
http://www.huduser.gov/portal/datasets/usps_crosswalk.html

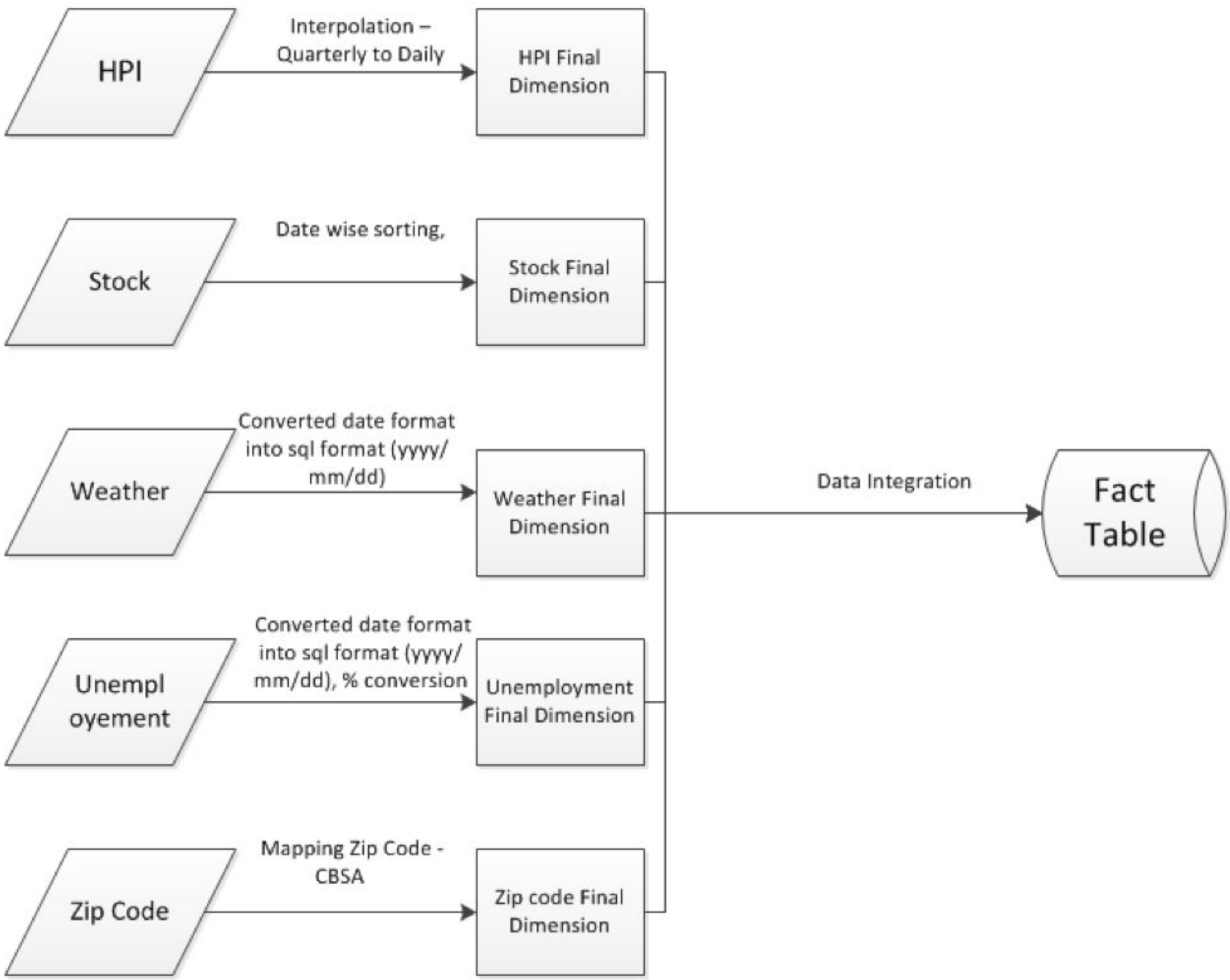
zip	cbsa
94503	34900
94508	34900
94515	34900
94558	34900
94559	34900
94562	34900
94567	34900
94573	34900

CBSA did not require cleansing, however it did require filtering by the CBSAs we were interested in. This was determined visually using this map:
http://www2.census.gov/geo/maps/metroarea/stcbsa_pg/Feb2013/cbsa2013_CA.pdf

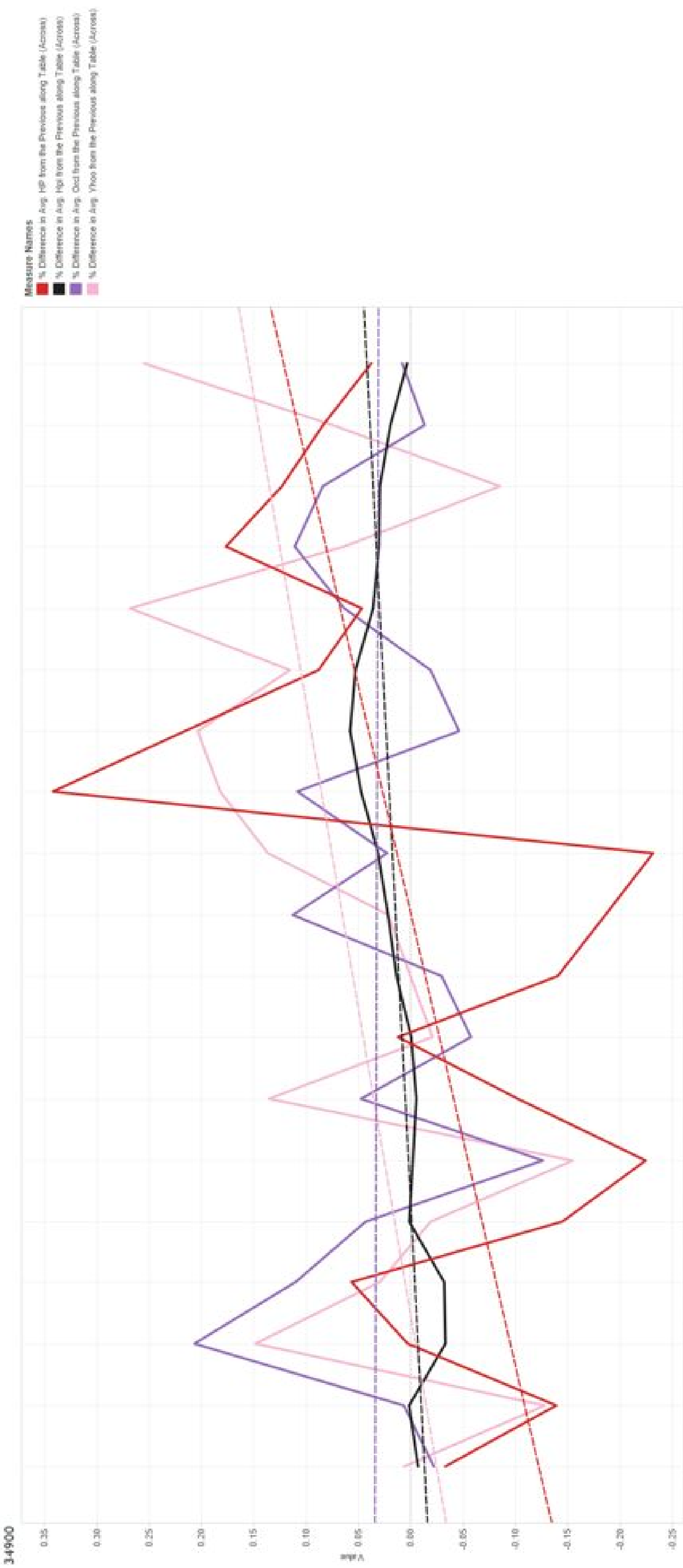
Dimensional Model:

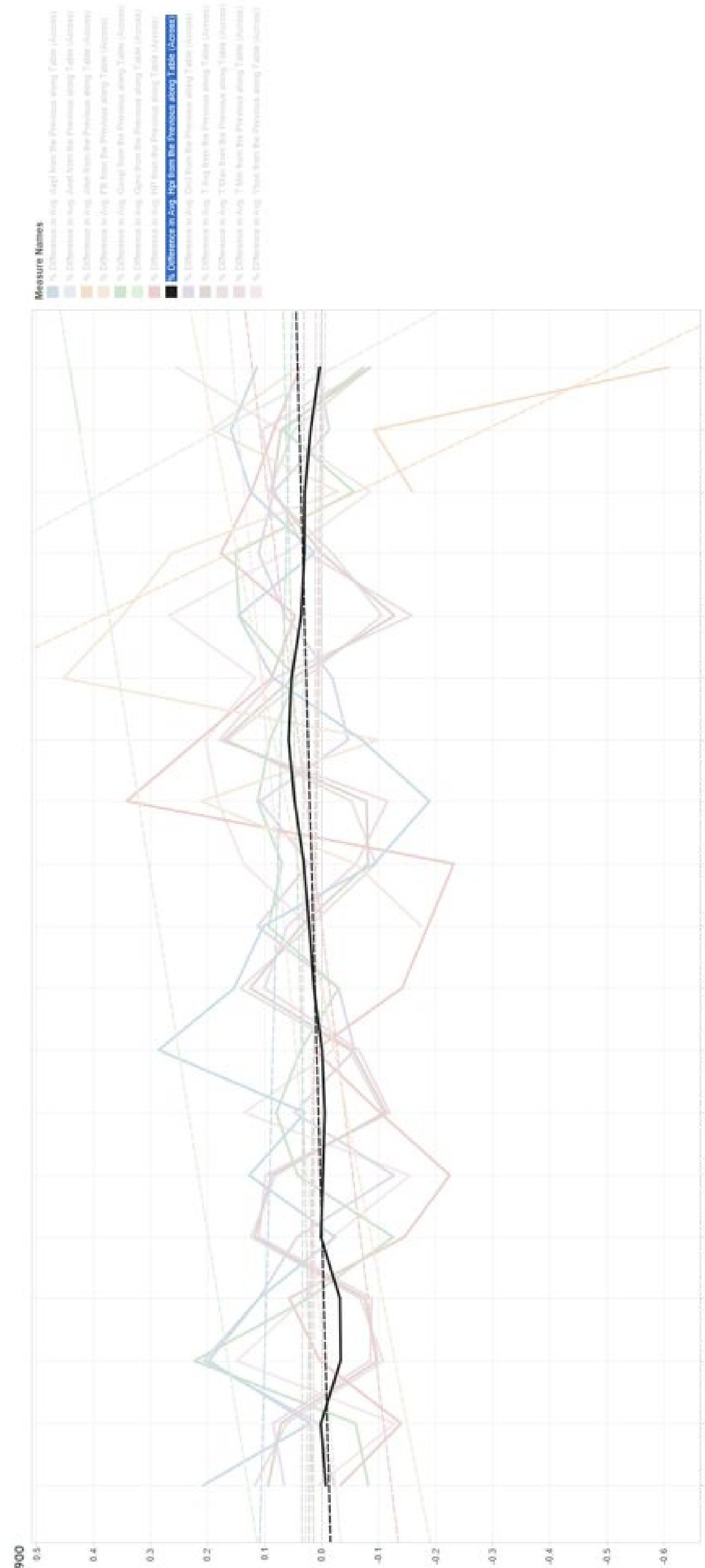


Data Integration Mapping

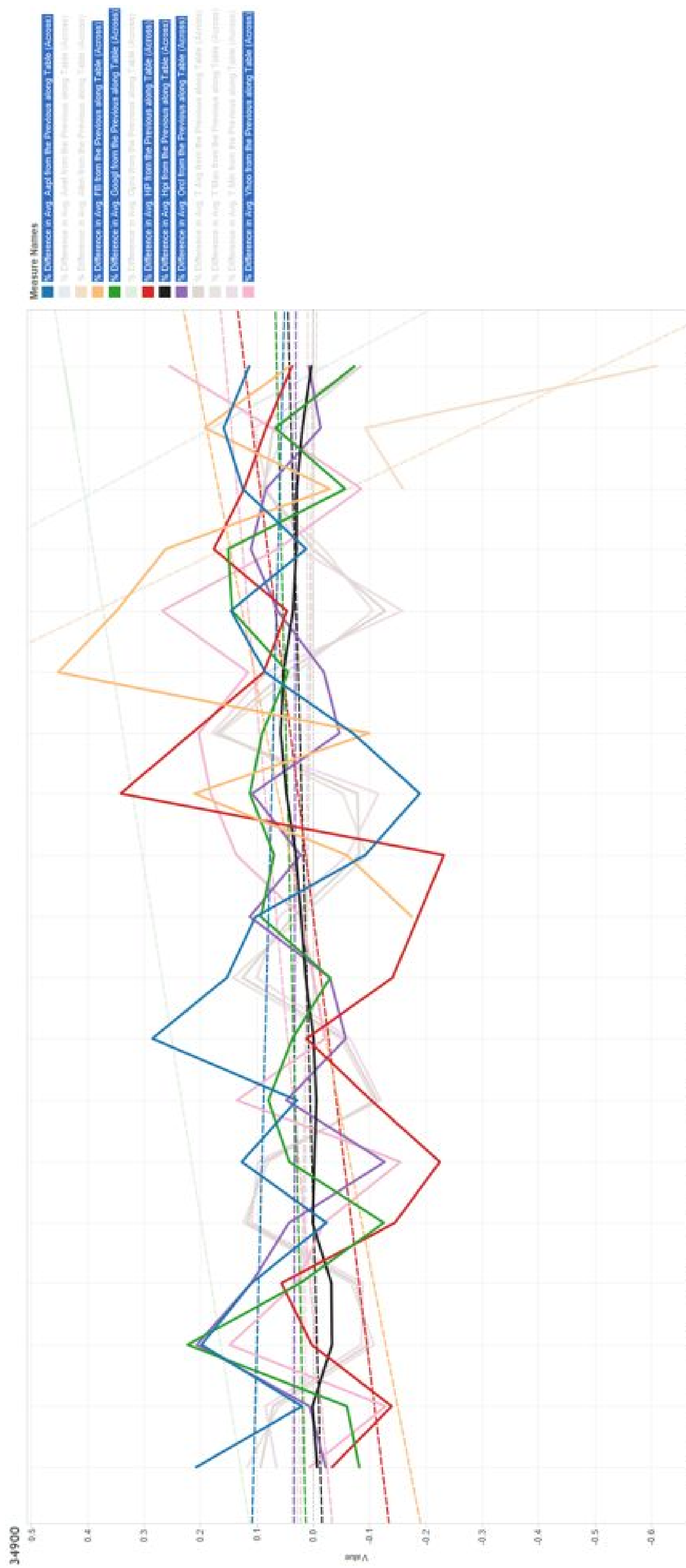


Printouts of Business Intelligence Technology:

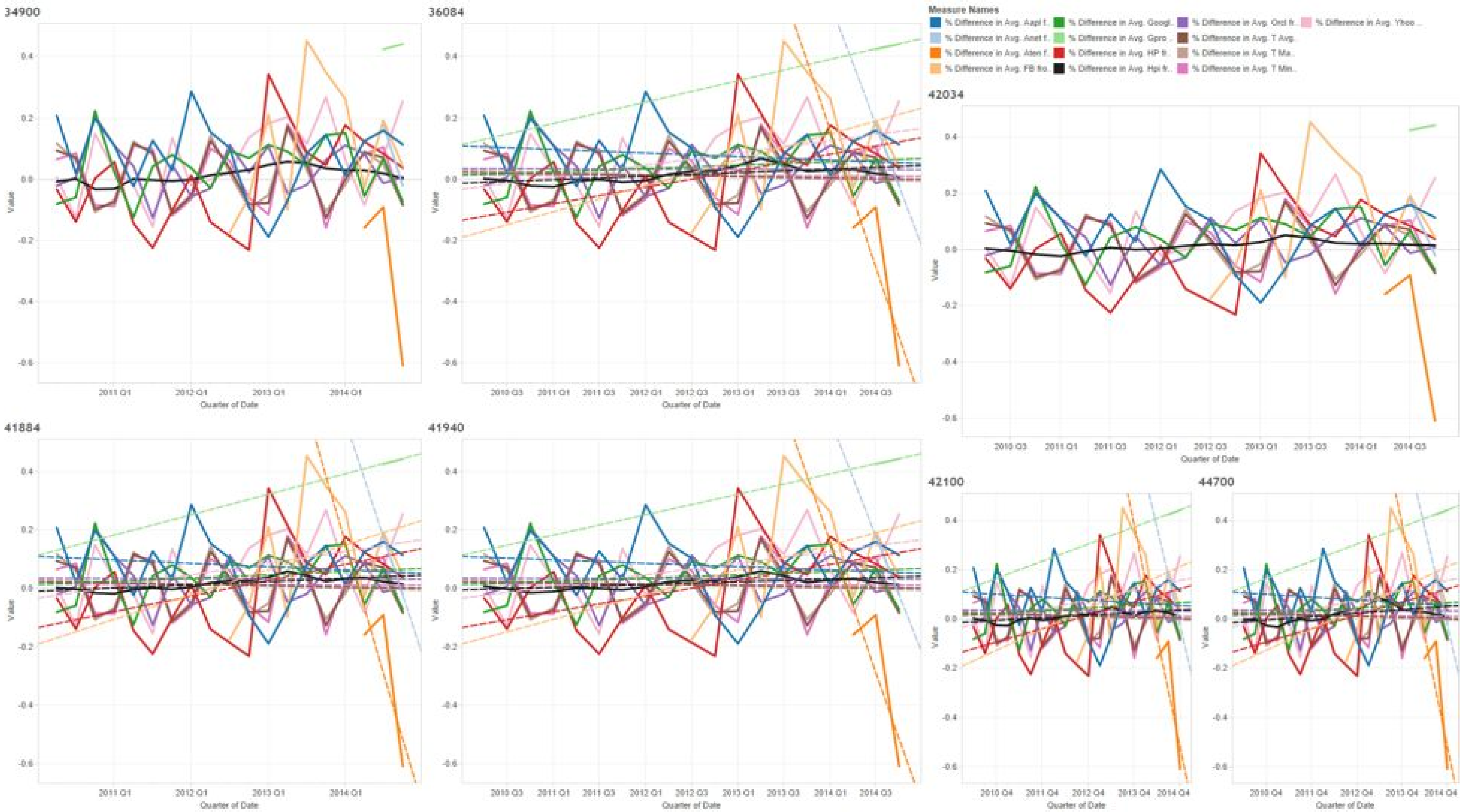




Measure Names
% Difference in Avg. Hpi from the Previous along Table (Across)
Quarter of Date
Value


34900
Measure Names
% Difference in Avg. Aapl from the Previous along Table (Across)
% Difference in Avg. Fb from the Previous along Table (Across)
% Difference in Avg. Googl from the Previous along Table (Across)
% Difference in Avg. HP from the Previous along Table (Across)
% Difference in Avg. Hpi from the Previous along Table (Across)
% Difference in Avg. Orcl from the Previous along Table (Across)
% Difference in Avg. Yhoo from the Previous along Table (Across)
Value
Quarter of Date
2010 Q2
2010 Q3
2010 Q4
2011 Q1
2011 Q2
2011 Q3
2011 Q4
2012 Q1
2012 Q2
2012 Q3
2012 Q4
2013 Q1
2013 Q2
2013 Q3
2013 Q4
2014 Q1
2014 Q2
2014 Q3
2014 Q4

Dashboard


34900
36084
41884
41940
42034
42100
44700
Value
Quarter of Date
Measure Names
% Difference in Avg. Aapl f..
% Difference in Avg. Anet f..
% Difference in Avg. Aten f..
% Difference in Avg. FB fro..
% Difference in Avg. Googl..
% Difference in Avg. Gpro ..
% Difference in Avg. HP fr..
% Difference in Avg. Hpi fr..
% Difference in Avg. Orcl fr..
% Difference in Avg. T Avg..
% Difference in Avg. T Ma..
% Difference in Avg. T Min..
% Difference in Avg. Yhoo ..

Source		Transformation	Target	Comment
Description: database name:			Description: database name:	
Table Name	**Column Name**	**Logic**	**Column Name**	
Facebook Stock Dimension	Price	finding the percentage change	Fact Table	
Google Stock Dimension	Price	finding the percentage change	Fact Table	
HP Stock Dimension	Price	finding the percentage change	Fact Table	
Yahoo Stock Dimension	Price	finding the percentage change	Fact Table	
Apple Stock Dimension	Price	finding the percentage change	Fact Table	
Oracle Stock Dimension	Price	finding the percentage change	Fact Table	
Anet Stock Dimension	Price	finding the percentage change	Fact Table	
Aten Stock Dimension	Price	finding the percentage change	Fact Table	
GoPro Stock	Price	finding the percentage change	Fact Table	
Weather		finding the percentage change	Fact Table	
Unemployment		finding the percentage change	Fact Table	
Housing Price Index		finding the percentage change	Fact Table	

TABLE_NAME [1]	TABLE_DESCRIPT [2]	TABLE_T [3]	SUBJECT_A [4]	DB_TYPE [5]	LOCATION [6]
Hpi	House Price Index	Dimension		MySQL	
Weather	Weather information	Dimension		MySQL	
School Ratings	School ratings information	Dimension		MySQL	
Crime Rate	Crime related data	Dimension		MySQL	
Unemployment Rate	Unemployment rate by CBSA	Dimension		MySQL	
Stock	Daily Adj closing price of each stocks per day	Dimension		MySQL	
CBSA	Mapping of all ZipCodes to CBSA Codes	Dimension		MySQL	

TABLE_NAME [1]	COL_NAME [2]	COL_BUS_NAME [3]	DESCRIPTION	DATA TYPE	NULLABLE Y/N	VALIDATION_R [4]	TRANSLATION_R [5]
Hpi	hpi_type		X	X			Set to NA if "NULL"
Hpi	hpi_flavor		X	X			Set to NA if "NULL"
Hpi	frequency		X	X			Set to NA if "NULL"
Hpi	level		X	X			Set to NA if "NULL"
Hpi	place_name		Traditional names.	Char			Set to NA if "NULL"
Hpi	place_id		Abbreviations or CBSA codes.	Int			Set to NA if "NULL"
Hpi	yr		Only the all-transactions data are published before 1991.	Int			Set to NA if "NULL"
Hpi	period		Period is either 1 through 4 for quarterly or 1 through 12 for monthly data.	Date			Set to NA if "NULL"
Hpi	index_nsa		index, non seasonally adjusted	Double			Set to NA if "NULL"
Hpi	index_sa		index, seasonally adjusted	Double			Set to NA if "NULL"
Weather	Date			date			Set to NA if "NULL"
Weather	PRCP (in)		Precipitation	double			Set to NA if "NULL"
Weather	TAVE (F)		Average Temperature	double			Set to NA if "NULL"
Weather	TMAX (F)		Maximum Temperature	double			Set to NA if "NULL"
Weather	TMIN (F)		Minimum Temperature	double			Set to NA if "NULL"
School Ratings	Zip code		This is the main point of reference for the table	Int			Set to NA if "NULL"
School Ratings	SNAME		Optional information that could lead to additonal analyses but not a part of the primary analysis	Character			Set to NA if "NULL"
School Ratings	DNAME		X	X			Set to NA if "NULL"
School Ratings	CNAME		X	X			Set to NA if "NULL"
School Ratings	AVG_NW		X	X			Set to NA if "NULL"
School Ratings	AVG_W		We wil be using this measurement for primary analysis/analyses.	Int			Set to NA if "NULL"
Crime Rate	Areaname		County names (X)	Character			Set to NA if "NULL"
Crime Rate	Year		Years from 1981 to 2014 (X)	Int			Set to NA if "NULL"
Crime Rate	Number of Crimes		Number of Violent crimes reported by DoJ-FBI(1981-2010) and by the Sheriff's office or county police department(2011-2014) (X)	Int			Set to NA if "NULL"
Unemployment Rate	#		X	X			Set to NA if "NULL"
Unemployment Rate	Zip Code		Zip codes of areas in the bay	Int			Set to NA if "NULL"
Unemployment Rate	Location		X	X			Set to NA if "NULL"
Unemployment Rate	City		Name of city	Char			Set to NA if "NULL"
Unemployment Rate	Population		X	X			Set to NA if "NULL"
Unemployment Rate	% Unemployment Rate		rate of Unemployment	Double			Set to NA if "NULL"
Unemployment Rate	National Rank		X	X			
Stock	Date		Date of the Adj Close price	Date			Set to NA if "NULL"
Stock	Adj Close		The adjusted closing price of the specific Stock on the given date	Float			Set to NA if "NULL"
ZipCode	Zip		ZipCode	Int			Set to NA if "NULL"
ZipCode	Name		X	X			Set to NA if "NULL"
ZipCode	empflag		X	X			Set to NA if "NULL"
ZipCode	emp_nf		X	X			Set to NA if "NULL"
ZipCode	emp		X	X			Set to NA if "NULL"
ZipCode	qp1_nf		X	X			Set to NA if "NULL"
ZipCode	ap_nf		X	X			Set to NA if "NULL"
ZipCode	est		X	X			Set to NA if "NULL"
ZipCode	city		X	X			Set to NA if "NULL"
ZipCode	stabbr		In the state of California	Varchar			Set to NA if "NULL"
ZipCode	cty_name		The name of the city in the State of California	Varchar			Set to NA if "NULL"

FILE_NAME [1]	FILE_DESCRIPTI [2]	SUBJECT A [3]	FILE_T [4]	LOCATI [5]	ONLINE_RETENT [6]	TOTAL_RO [7]	RUN_FREQUE [8]	EXTRACT_AVAILA [9]	MAX_ROW_ [10]	VALIDATION_R [11]	PUSHED_OR_PU [12]	RECEIPT_ACK_REQUI [13]	EXTRACT_READINESS_INDIC [14]	SOURCE_DOCUMENTA [15]
ANET.csv	Daily Stock adj Closing price for the recent IPO companies in the bay area	Stock	.csv	finance.yahoo.com		352	Daily		352	NA	PULLED	NO	NA	NA
ATEN.csv	Daily Stock adj Closing price for the recent IPO companies in the bay area	Stock	.csv	finance.yahoo.com		405	Daily		405	NA	PULLED	NO	NA	NA
GoPro.csv	Daily Stock adj Closing price for the recent IPO companies in the bay area	Stock	.csv	finance.yahoo.com		338	Daily		338	NA	PULLED	NO	NA	NA
FIT.csv	Daily Stock adj Closing price for the recent IPO companies in the bay area	Stock	.csv	finance.yahoo.com		92	Daily		92	NA	PULLED	NO	NA	NA
yhoo.csv	Daily Stock adj closing price for bay area tech companies	Stock	.csv	finance.yahoo.com		1467	Daily		1467	NA	PULLED	NO	NA	NA
oracle.csv	Daily Stock adj closing price for bay area tech companies	Stock	.csv	finance.yahoo.com		1467	Daily		1467	NA	PULLED	NO	NA	NA
hp.csv	Daily Stock adj closing price for bay area tech companies	Stock	.csv	finance.yahoo.com		1467	Daily		1467	NA	PULLED	NO	NA	NA
googl.csv	Daily Stock adj closing price for bay area tech companies	Stock	.csv	finance.yahoo.com		1467	Daily		1467	NA	PULLED	NO	NA	NA
fb.csv	Daily Stock adj closing price for bay area tech companies	Stock	.csv	finance.yahoo.com		868	Daily		868	NA	PULLED	NO	NA	NA
aapl.csv	Daily Stock adj closing price for bay area tech companies	Stock	.csv	finance.yahoo.com		1467	Daily		1467	NA	PULLED	NO	NA	NA
All_hpi.csv	CBSA ID, HPI, Place Name, and the Date	HPI	.csv			37059	Quarterly		37059	NA	PULLED	NO	NA	NA
stockton_44700.csv	CBSA ID, HPI, Place Name, and the Date	HPI	.csv			5295	Quarterly		5295	NA	PULLED	NO	NA	NA
sf_41884.csv	CBSA ID, HPI, Place Name, and the Date	HPI	.csv			5295	Quarterly		5295	NA	PULLED	NO	NA	NA
santacruz_42100.csv	CBSA ID, HPI, Place Name, and the Date	HPI	.csv			5295	Quarterly		5295	NA	PULLED	NO	NA	NA
sanrafael_42034.csv	CBSA ID, HPI, Place Name, and the Date	HPI	.csv			5295	Quarterly		5295	NA	PULLED	NO	NA	NA
sanjose_41940.csv	CBSA ID, HPI, Place Name, and the Date	HPI	.csv			5295	Quarterly		5295	NA	PULLED	NO	NA	NA
oakland_36084.csv	CBSA ID, HPI, Place Name, and the Date	HPI	.csv			5295	Quarterly		5295	NA	PULLED	NO	NA	NA
napa_34900.csv	CBSA ID, HPI, Place Name, and the Date	HPI	.csv			5295	Quarterly		5295	NA	PULLED	NO	NA	NA
Berkley Weather.csv	Daily Precipitation, Average Temp, Max Temp, and Min Temp info for the bay area as a unit	Weather	.csv	**http://cdiac.ornl.gov/epubs/ndp/ http://cdiac.ornl.gov/cgi-bin/bro**		5418	Daily		5418	NA	PULLED	NO	NA	NA
bay_area_zip_cbsa.csv	Mapping of the Zipcode data to the CBSA data for all Cities in CA	Zip Code / CBSA	.csv	http://maps.huge.info/zip.htm https://www.census.gov/econ/cbp/		416	FIXED		416	NA	PULLED	NO	NA	NA

FILE_NA [1]	COL_NAME [2]	COL_BUS_NAME [3]	DESCRIPTION	DATA TYPE	NULLABLE Y/N?	VALIDATION_R [4]	TRANSLATION_R [5]
ANET.csv	Date	Same as Col_Name	Date of the closing price	Date	Y	NA	Put NA if Null
	Adj Close	Same as Col_Name	Adjusted closin	float	Y	NA	Put NA if Null
ATEN.csv	Date	Same as Col_Name	Date of the closing price	Date	Y	NA	Put NA if Null
	Adj Close	Same as Col_Name	Adjusted closin	float	Y	NA	Put NA if Null
GoPro.csv	Date	Same as Col_Name	Date of the closing price	Date	Y	NA	Put NA if Null
	Adj Close	Same as Col_Name	Adjusted closin	float	Y	NA	Put NA if Null
FIT.csv	Date	Same as Col_Name	Date of the closing price	Date	Y	NA	Put NA if Null
	Adj Close	Same as Col_Name	Adjusted closin	float	Y	NA	Put NA if Null
yhoo.csv	Date	Same as Col_Name	Date of the closing price	Date	Y	NA	Put NA if Null
	Adj Close	Same as Col_Name	Adjusted closin	float	Y	NA	Put NA if Null
oracle.csv	Date	Same as Col_Name	Date of the closing price	Date	Y	NA	Put NA if Null
	Adj Close	Same as Col_Name	Adjusted closin	float	Y	NA	Put NA if Null
hp.csv	Date	Same as Col_Name	Date of the closing price	Date	Y	NA	Put NA if Null
	Adj Close	Same as Col_Name	Adjusted closin	float	Y	NA	Put NA if Null
googl.csv	Date	Same as Col_Name	Date of the closing price	Date	Y	NA	Put NA if Null
	Adj Close	Same as Col_Name	Adjusted closin	float	Y	NA	Put NA if Null
fb.csv	Date	Same as Col_Name	Date of the closing price	Date	Y	NA	Put NA if Null
	Adj Close	Same as Col_Name	Adjusted closin	float	Y	NA	Put NA if Null
aapl.csv	Date	Same as Col_Name	Date of the closing price	Date	Y	NA	Put NA if Null
	Adj Close	Same as Col_Name	Adjusted closin	float	Y	NA	Put NA if Null
All_hpi.csv	Date	Same as Col_Name	Date	Date		NA	Put NA if Null
	place_id	Same as Col_Name	CBSA code	Int		NA	Put NA if Null
	place_name	Same as Col_Name	Name of the City	varchar		NA	Put NA if Null
	hpi	Same as Col_Name	HPI number at this given date and city	float		NA	Put NA if Null
stockton_44700.csv	Date	Same as Col_Name	Date	Date		NA	Put NA if Null
	place_id	Same as Col_Name	CBSA code	Int		NA	Put NA if Null
	place_name	Same as Col_Name	Name of the City	varchar		NA	Put NA if Null
	hpi	Same as Col_Name	HPI number at this given date and city	float		NA	Put NA if Null

sf_41884.csv	Date	Same as Col_Name	Date	Date		NA	Put NA if Null
	place_id	Same as Col_Name	CBSA code	Int		NA	Put NA if Null
	place_name	Same as Col_Name	Name of the City	varchar		NA	Put NA if Null
	hpi	Same as Col_Name	HPI number at this given date and city	float		NA	Put NA if Null
santacruz_42100.csv	Date	Same as Col_Name	Date	Date		NA	Put NA if Null
	place_id	Same as Col_Name	CBSA code	Int		NA	Put NA if Null
	place_name	Same as Col_Name	Name of the City	varchar		NA	Put NA if Null
	hpi	Same as Col_Name	HPI number at this given date and city	float		NA	Put NA if Null
sanrafael_42034.csv	Date	Same as Col_Name	Date	Date		NA	Put NA if Null
	place_id	Same as Col_Name	CBSA code	Int		NA	Put NA if Null
	place_name	Same as Col_Name	Name of the City	varchar		NA	Put NA if Null
	hpi	Same as Col_Name	HPI number at this given date and city	float		NA	Put NA if Null
sanjose_41940.csv	Date	Same as Col_Name	Date	Date		NA	Put NA if Null
	place_id	Same as Col_Name	CBSA code	Int		NA	Put NA if Null
	place_name	Same as Col_Name	Name of the City	varchar		NA	Put NA if Null
	hpi	Same as Col_Name	HPI number at this given date and city	float		NA	Put NA if Null
oakland_36084.csv	Date	Same as Col_Name	Date	Date		NA	Put NA if Null
	place_id	Same as Col_Name	CBSA code	Int		NA	Put NA if Null
	place_name	Same as Col_Name	Name of the City	varchar		NA	Put NA if Null
	hpi	Same as Col_Name	HPI number at this given date and city	float		NA	Put NA if Null
napa_34900.csv	Date	Same as Col_Name	Date	Date		NA	Put NA if Null
	place_id	Same as Col_Name	CBSA code	Int		NA	Put NA if Null

	place_name	Same as Col_Name	Name of the City	varchar		NA	Put NA if Null
	hpi	Same as Col_Name	HPI number at this given date and city	float		NA	Put NA if Null
Berkley Weather.csv	Date	Same as Col_Name	Date	Date		NA	Put NA if Null
	PRCP (in)	Precipitation in Inches	Amount of Rain	float		NA	Put NA if Null
	TAVE (F)	Ave Temp in F	average tempe	int		NA	Put NA if Null
	TMAX (F)	Max Temp in F	Max temperatu	int		NA	Put NA if Null
	TMIN (F)	Min Temp in F	Min temperatur	int		NA	Put NA if Null
bay_area_zip_cbsa.csv	Zip	Same as Col_Name	Zip Code of the	int		NA	Put NA if Null
	CBSA	Same as Col_Name	corresponding	int		NA	Put NA if Null
	name	Same as Col_Name	name of the cit	varchar		NA	Put NA if Null
	city	Same as Col_Name	name of the cit	varchar		NA	Put NA if Null
	cty_name	Same as Col_Name	name of the cit	varchar		NA	Put NA if Null

Analysis of Variance:					
Field	DF	SSE	MSE	F	p-value
Measure Names	24	1.0962841	0.0456785	3.77968	< 0.0001
Individual trend lines:					
Color	Coefficients				
Measure Names	p-value	StdErr	t-value	p-value	
HPI	0.0017698	9.05E-06	3.70209	0.0017698	
HP	0.0346872	6.46E-05	2.29564	0.0346872	
Yhoo	0.0449936	5.06E-05	2.16399	0.0449936	
FB	0.387697	0.0002542	0.913468	0.387697	
Aten	0.407646	0.0018354	-1.34218	0.407646	
Googl	0.492979	4.25E-05	0.700685	0.492979	
Aapl	0.566065	5.41E-05	-0.585254	0.566065	
T Max	0.666484	4.39E-05	-0.438586	0.666484	
T Avg	0.790266	4.47E-05	-0.270191	0.790266	
Orcl	0.958986	3.71E-05	-0.0521896	0.958986	
T Min	0.962417	4.78E-05	-0.04782	0.962417	
Gpro	N/A	Since the trend line model has zero residual degr			
Anet	N/A	Since the trend line model has zero residual degr			

Napa

Analysis of Variance:					
Field	DF	SSE	MSE	F	p-value
Measure Names	24	1.0957841	0.0456577	3.77872	< 0.0001
Individual trend lines:					
Color	Coefficients				
Measure Names	p-value	StdErr	t-value	p-value	
HPI	0.0021248	8.78E-06	3.61788	0.0021248	
HP	0.0346872	6.46E-05	2.29564	0.0346872	
Yhoo	0.0449936	5.06E-05	2.16399	0.0449936	
FB	0.387697	0.0002542	0.913468	0.387697	
Aten	0.407646	0.0018354	-1.34218	0.407646	
Googl	0.492979	4.25E-05	0.700685	0.492979	
Aapl	0.566065	5.41E-05	-0.585254	0.566065	
T Max	0.666484	4.39E-05	-0.438586	0.666484	
T Avg	0.790266	4.47E-05	-0.270191	0.790266	
Orcl	0.958986	3.71E-05	-0.0521896	0.958986	
T Min	0.962417	4.78E-05	-0.04782	0.962417	
Gpro	N/A	Since the trend line model has zero residual degrees o			
Anet	N/A	Since the trend line model has zero residual degrees o			

Oakland

Analysis of Variance:					
Field	DF	SSE	MSE	F	p-value
Measure Names	24	1.0947873	0.0456161	3.78032	< 0.0001
Individual trend lines:					
Color	Coefficients				
Measure Names	p-value	StdErr	t-value	p-value	
HPI	0.0003493	6.69E-06	4.45281	0.0003493	
HP	0.0346872	6.46E-05	2.29564	0.0346872	
Yhoo	0.0449936	5.06E-05	2.16399	0.0449936	
FB	0.387697	0.0002542	0.913468	0.387697	
Aten	0.407646	0.0018354	-1.34218	0.407646	
Googl	0.492979	4.25E-05	0.700685	0.492979	
Aapl	0.566065	5.41E-05	-0.585254	0.566065	
T Max	0.666484	4.39E-05	-0.438586	0.666484	
T Avg	0.790266	4.47E-05	-0.270191	0.790266	
Orcl	0.958986	3.71E-05	-0.0521896	0.958986	
T Min	0.962417	4.78E-05	-0.04782	0.962417	
Gpro	N/A	Since the trend line model has zero residual degree			
Anet	N/A	Since the trend line model has zero residual degree			

San Francisco

Analysis of Variance:					
Field	DF	SSE	MSE	F	p-value
Measure Names	24	1.0948898	0.0456204	3.77997	< 0.0001
Individual trend lines:					
Color	Coefficients				
Measure Names	p-value	StdErr	t-value	p-value	
HPI	0.0021331	7.02E-06	3.6161	0.0021331	
HP	0.0346872	6.46E-05	2.29564	0.0346872	
Yhoo	0.0449936	5.06E-05	2.16399	0.0449936	
FB	0.387697	0.0002542	0.913468	0.387697	
Aten	0.407646	0.0018354	-1.34218	0.407646	
Googl	0.492979	4.25E-05	0.700685	0.492979	
Aapl	0.566065	5.41E-05	-0.585254	0.566065	
T Max	0.666484	4.39E-05	-0.438586	0.666484	
T Avg	0.790266	4.47E-05	-0.270191	0.790266	
Orcl	0.958986	3.71E-05	-0.0521896	0.958986	
T Min	0.962417	4.78E-05	-0.04782	0.962417	
Gpro	N/A	Since the trend line model has zero residual degrees of fr			
Anet	N/A	Since the trend line model has zero residual degrees of fr			

San Jose

Analysis of Variance:					
Field	DF	SSE	MSE	F	p-value
Measure Names	24	1.0983122	0.045763	3.79349	< 0.0001
Individual trend lines:					
Color	Coefficients				
Measure Names	p-value	StdErr	t-value	p-value	
HPI	0.0008295	6.20E-06	4.05142	0.0008295	
HP	0.0346872	6.46E-05	2.29564	0.0346872	
Yhoo	0.0449936	5.06E-05	2.16399	0.0449936	
FB	0.387697	0.0002542	0.913468	0.387697	
Aten	0.407646	0.0018354	-1.34218	0.407646	
Googl	0.492979	4.25E-05	0.700685	0.492979	
Aapl	0.566065	5.41E-05	-0.585254	0.566065	
T Max	0.666484	4.39E-05	-0.438586	0.666484	
T Avg	0.790266	4.47E-05	-0.270191	0.790266	
Orcl	0.958986	3.71E-05	-0.0521896	0.958986	
T Min	0.962417	4.78E-05	-0.04782	0.962417	
Gpro	N/A	Since the trend line model has zero residual degre			
Anet	N/A	Since the trend line model has zero residual degre			

Santa Rafael

Analysis of Variance:					
Field	DF	SSE	MSE	F	p-value
Measure Names	24	1.0990698	0.0457946	3.79581	< 0.0001
Individual trend lines:					
Color	Coefficients				
Measure Names	p-value	StdErr	t-value	p-value	
HPI	0.0003447	6.35E-06	4.45901	0.0003447	
HP	0.0346872	6.46E-05	2.29564	0.0346872	
Yhoo	0.0449936	5.06E-05	2.16399	0.0449936	
FB	0.387697	0.0002542	0.913468	0.387697	
Aten	0.407646	0.0018354	-1.34218	0.407646	
Googl	0.492979	4.25E-05	0.700685	0.492979	
Aapl	0.566065	5.41E-05	-0.585254	0.566065	
T Max	0.666484	4.39E-05	-0.438586	0.666484	
T Avg	0.790266	4.47E-05	-0.270191	0.790266	
Orcl	0.958986	3.71E-05	-0.0521896	0.958986	
T Min	0.962417	4.78E-05	-0.04782	0.962417	
Gpro	N/A	Since the trend line model has zero residual degrees of			
Anet	N/A	Since the trend line model has zero residual degrees of			

Santa Cruz

Analysis of Variance:					
Field	DF	SSE	MSE	F	p-value
Measure Names	24	1.0939957	0.045583	3.76291	< 0.0001
Individual trend lines:					
Color	Coefficients				
Measure Names	p-value	StdErr	t-value	p-value	
HPI	0.005148	1.18E-05	3.20891	0.005148	
HP	0.034687	6.46E-05	2.29564	0.034687	
Yhoo	0.044994	5.06E-05	2.16399	0.044994	
FB	0.387697	0.000254	0.913468	0.387697	
Aten	0.407646	0.001835	-1.34218	0.407646	
Googl	0.492979	4.25E-05	0.700685	0.492979	
Aapl	0.566065	5.41E-05	-0.58525	0.566065	
T Max	0.666484	4.39E-05	-0.43859	0.666484	
T Avg	0.790266	4.47E-05	-0.27019	0.790266	
Orcl	0.958986	3.71E-05	-0.05219	0.958986	
T Min	0.962417	4.78E-05	-0.04782	0.962417	
Gpro	N/A	Since the trend line model has zero residual degrees			
Anet	N/A	Since the trend line model has zero residual degrees			

Stockton

```python
import csv
import datetime
import sys

file = sys.argv[1]

newfile = []

with open(file, 'rb') as csvfile:
    readfile = csv.reader(csvfile)
    for row in readfile:
        if row[0] == "Date":
            newfile.append(row)
        else:
            date = row[0]
            dateobj = datetime.datetime.strptime(date, "%m/%d/%Y")
            newdate = dateobj.strftime("%Y-%m-%d")

            newfile.append([newdate,row[1]])

with open("new/"+file, "w") as output:
    writer = csv.writer(output, lineterminator='\n')
    writer.writerows(newfile)
```

```
import csv
import calendar
import datetime

csvfile = []
newcsv = []

reader = csv.reader(open("extended/vallejo_extended.csv"))
for row in reader:
    csvfile.append(row)

header = csvfile.pop(0)
#print header

newheader = ["date","place_id","place_name","hpi"]


for row in csvfile:
    cal = calendar.Calendar()
    numdays = []
    daysinmonth = cal.itermonthdays(int(row[0]),int(row[1]))
    for each in daysinmonth:
        if each > 0:
            numdays.append(each)

    for each in numdays:
        eachdate = datetime.date(int(row[0]),int(row[1]), each)
        formatteddate = eachdate.strftime("%m/%d/%y")

        if each == 1:
            newcsv.append([formatteddate,row[2],row[3],row[4]])
        else:
            newcsv.append([formatteddate,row[2],row[3],""])

with open("blank/vallejo_blank.csv", "w") as output:
    writer = csv.writer(output, lineterminator='\n')
    writer.writerow(newheader)
    writer.writerows(newcsv)
```

```python
import csv
import calendar

csvfile = []
newcsv = []

reader = csv.reader(open("orig/vallejo.csv"))
for row in reader:
    csvfile.append(row)

header = csvfile.pop(0)
#print header

for row in csvfile:
    if row[1] == "1":
        newcsv.append([row[0],"1",row[2],row[3],row[4]])
        newcsv.append([row[0],"2",row[2],row[3],""])
        newcsv.append([row[0],"3",row[2],row[3],""])
    elif row[1] == "2":
        newcsv.append([row[0],"4",row[2],row[3],row[4]])
        newcsv.append([row[0],"5",row[2],row[3],""])
        newcsv.append([row[0],"6",row[2],row[3],""])
    elif row[1] == "3":
        newcsv.append([row[0],"7",row[2],row[3],row[4]])
        newcsv.append([row[0],"8",row[2],row[3],""])
        newcsv.append([row[0],"9",row[2],row[3],""])
    elif row[1] == "4":
        newcsv.append([row[0],"10",row[2],row[3],row[4]])
        newcsv.append([row[0],"11",row[2],row[3],""])
        newcsv.append([row[0],"12",row[2],row[3],""])

with open("extended/vallejo_extended.csv", "w") as output:
    writer = csv.writer(output, lineterminator='\n')
    writer.writerow(header)
    writer.writerows(newcsv)
```

```
import csv

dict = {}

with open('school.csv', 'rb') as csvfile:
    reader = csv.reader(csvfile)
    for row in reader:
        if row[1] == '':
            pass
        else:
            key = int(row[0])
            val = int(row[1])
            if key not in dict:
                dict[key] = [val]
            else:
                dict[key].append(val)
keysindict = dict.keys()
ziplist = []
for each in keysindict:
    calc = sum(dict[each])/len(dict[each])
    ziplist.append([each,calc])

with open("results.csv", "w") as output:
    writer = csv.writer(output, lineterminator='\n')
    writer.writerows(ziplist)
```

What Affects Housing Prices in the Bay Area?

•••

Team **Kickass**

Bhakti Mohadkar, Frederick Su, Mike Greco, Sagar Botta, Sangramsingh Kardekar

Project Description

Macroeconomic variables like the stock market and housing price are highly interconnected. These relationships can be identified with Business Intelligence.

Regional influences can be anticipated by identifying major employers, significant environmental variables, and regional events of interest.

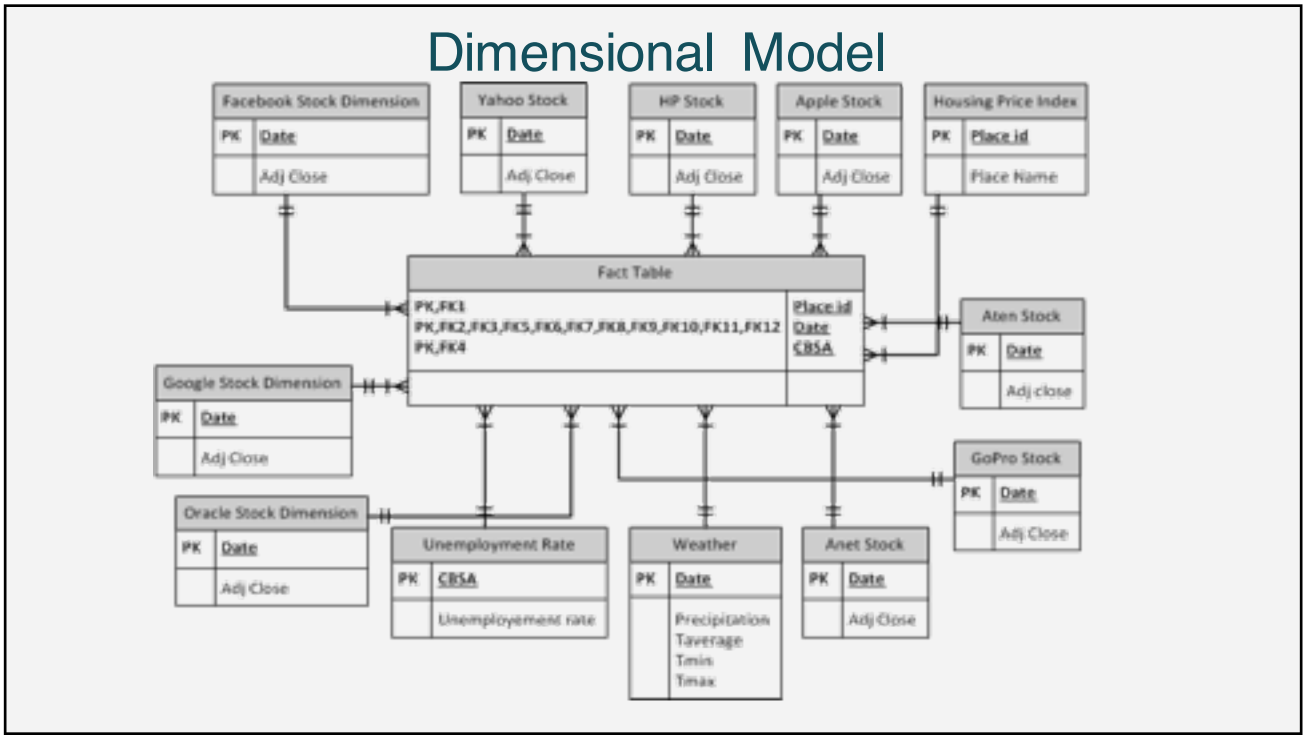
We aim to determine how variations in these factors can influence the average price of housing in the San Francisco Bay Area. This will allow a savvy investor to predict market trends before they happen, not while they are already in play.

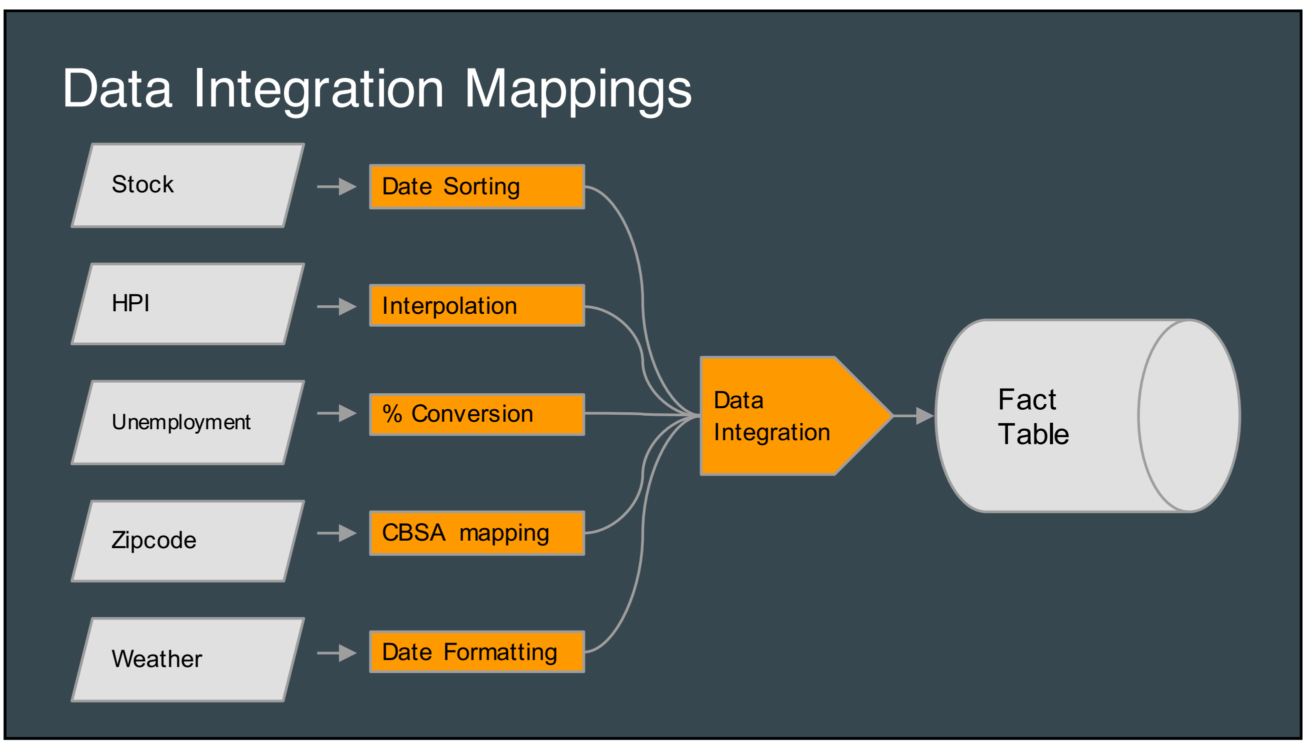
Team Member Responsibilities

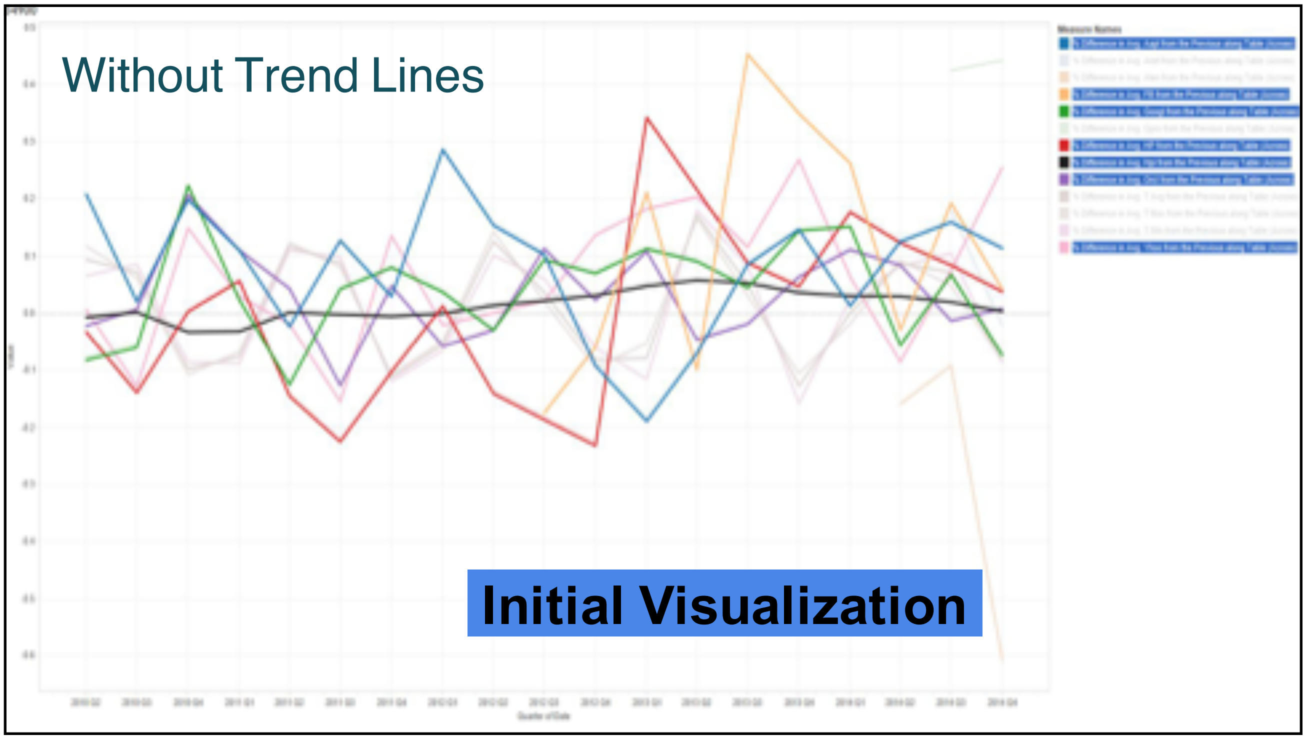
Bhakti Mohadkar *Data Integration Expert*	Identify usable data sources. Maintained data guidelines and data dictionary. Data Guardian.
Frederick Su *Business Analyst*	Identify usable data sources, converting high-level requirements into quantifiable data. Ensuring the implementation matches the concept.
Mike Greco *Project Manager*	Bringing together all aspects of the project. Filling in the gaps and verifying correctness at each stage. Resident do-it-all.
Sagar Botta *Software Engineer*	Automating data cleansing and integration. Manipulating data sources and deriving insights from visuals. Developer.
Sangramsingh Kardekar *Technical Architect*	Identified viable tools, ensuring a smooth interface between the technical stages of the project.

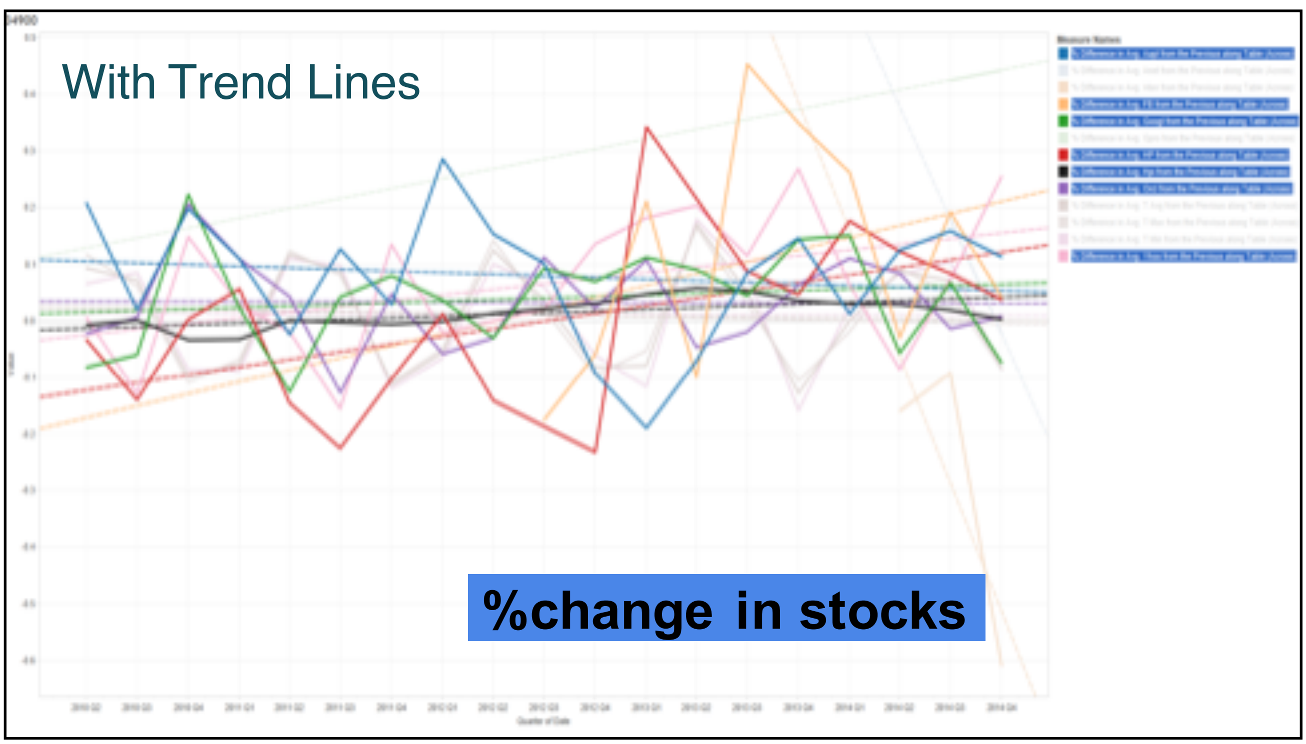
Data Sources and Issues

HPI (Housing Price Index)	- Federal Housing Finance Agency. Weighted index without unit.
Stocks	- Yahoo Finance. Identification of key regional influencers.
Weather	- US Historical Climatology Network. Limited Regional Scope.
Unemployment	- ZipAtlas.com. No numerical identifier (ZIP code)
Location Mapping	- US Census and US HUD. Multi-dimensional mapping required.


Dimensional Model
Facebook Stock Dimension
PK Date
Adj Close
Yahoo Stock
PK Date
Adj Close
HP Stock
PK Date
Adj Close
Apple Stock
PK Date
Adj Close
Housing Price Index
PK Place id
Place Name
Fact Table
PK,FK1
PK,FK2,FK3,FK5,FK6,FK7,FK8,FK9,FK10,FK11,FK12
PK,FK4
Place id
Date
CBSA
Aten Stock
PK Date
Adj close
Google Stock Dimension
PK Date
Adj Close
GoPro Stock
PK Date
Adj Close
Oracle Stock Dimension
PK Date
Adj Close
Unemployment Rate
PK CBSA
Unemployement rate
Weather
PK Date
Precipitation
Taverage
Tmin
Tmax
Anet Stock
PK Date
Adj Close


Data Integration Mappings
Stock
HPI
Unemployment
Zipcode
Weather
Date Sorting
Interpolation
% Conversion
CBSA mapping
Date Formatting
Data Integration
Fact Table


Without Trend Lines
Initial Visualization


With Trend Lines
%change in stocks

Statistical Analysis - Analysis of Variance

- A lower p- value indicates a greater influence within the model.
- A low overall p-value indicates the model is a good fit.

Analysis of Variance:					
Field	DF	SSE	MSE	F	p-value
Measure Names	24	1.0962841	0.0456785	3.77968	< 0.0001
Individual trend lines:					
Color	Coefficients				
Measure Names	p-value	StdErr	t-value	p-value	
HP	0.0346872	6.46E-05	2.29564	0.0346872	
Yhoo	0.0449936	5.06E-05	2.16399	0.0449936	
FB	0.387697	0.0002542	0.913468	0.387697	
Aten	0.407646	0.0018354	-1.34218	0.407646	
Googl	0.492979	4.25E-05	0.700685	0.492979	
Aapl	0.566065	5.41E-05	-0.585254	0.566065	
T Max	0.666484	4.39E-05	-0.438586	0.666484	
T Avg	0.790266	4.47E-05	-0.270191	0.790266	
Ord	0.958986	3.71E-05	-0.0521896	0.958986	
T Min	0.962417	4.78E-05	-0.04782	0.962417	
Gpro	N/A	Since the trend line model has zero residual degre			
Anet	N/A	Since the trend line model has zero residual degre			

Observations and Approximations

Housing price correlates closely with the stock prices of HP Inc. and Yahoo!

Housing prices across the Bay Area trend closely with each other.

Startup IPOs do not have measurable impact on average housing price.

If you are looking to buy or sell, watch HPQ & YHOO closely!

What We Learned

- Data transformation is more time consuming than we thought!
- Data is not always usable, especially if it was free.
- Having a well defined plan can save lots of time.
- Business Analytics software is extremely powerful, but isn't always perfect.
- It is more performant to store calculations in the data warehouse than performing them on-demand at the analytics stage.

Thank You.