Notes

Basic analysis results on dataset http://archive.ics.uci.edu/ml/datasets/Online+Retail+II

Basic python script https://github.com/nicola-orlando/tensorflow/tree/master/simple_tutorials/simple_customers_analysis

Dataset basic information

- Dataset http://archive.ics.uci.edu/ml/datasets/Online+Retail+II
  - For convenience, but not necessary, I split this into 2010 and 2011 data, I focus on the latter, extension to the full dataset is straightforward
- Other assumptions
  - Will not use full time (InvoiceDate) information, trimmed out
  - Remove entries corresponding to NANs (usually happens when incomplete data is stored)
  - Always assume that users buy a product and don't return it (some entries have negative valued Quantity field)

```
# For reference, starting header will look like this
#['Invoice' 'StockCode' 'Description' 'Quantity' 'InvoiceDate' 'Price' 'Customer ID' 'Country']

# Load data
# Need to enforce encoding as described here https://stackoverflow.com/questions/18171739/unicodedecodeerror-when-reading-csv-f
df = pd.read_csv('online_retail_II_2011.csv', engine='python')
# Here I want to clean up some information from the InvoiceDate column (don't plan to use time and year, just day and month)
df['InvoiceDate'] = df['InvoiceDate'].str.slice(3, -6)

print("Prining head of the file to see how it looks")
print(df.head())
print("Prining data types")
print(df.dtypes)

# Very first step, remove lines with incomplete data (e.g. missing Customer IDs).
df = df.dropna()
```

Generic treatment and basic features

- Manipulate data based on dataframe functionalities (grouping features, averages, sums, ..)
- Plot the results obtained in this way with matlib plot

Examples of operations on dataframes

```
def get_grouped_sum_multiplied(dataframe,grouping_feature,manipulated_data,manipulated_data_second):
    dataframe['ValueM'] = manipulated_data.abs() * manipulated_data_second.abs()
    grouped_data = dataframe.groupby(grouping_feature)['ValueM'].sum().reset_index(name='ValueM')
    return grouped_data
```

Used to calculate total cost of purchases per invoice and group (and more)

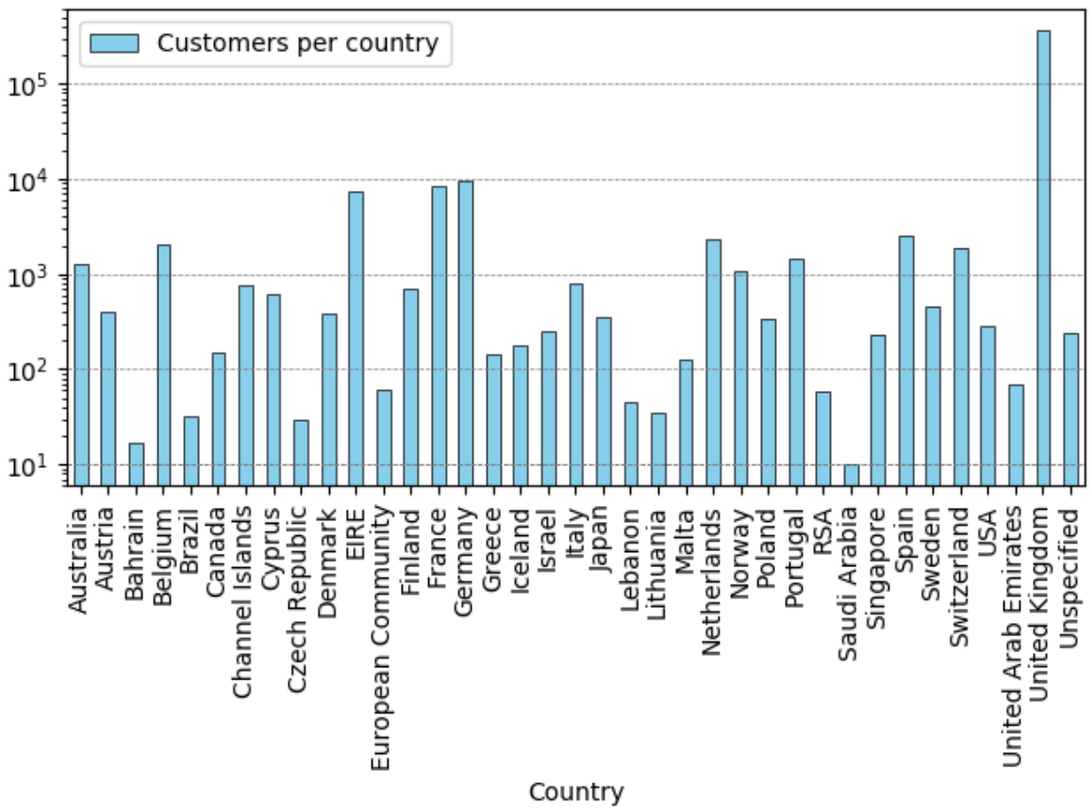
```
def get_counting(dataframe,grouping_feature,manipulated_feature,title='count'):
    grouped_data = dataframe.groupby(grouping_feature)[manipulated_feature].count().reset_index(name=title)
    return grouped_data
```

Used to total number of customers per country (and more)

```
def make_chart_plot(dataframe,x_axis_name,do_log_y,lines_coord,plot_title,plot_kind):
    dataframe.plot(kind=plot_kind,x=x_axis_name,logy=do_log_y)
    plt.axhline(y=10,color='gray',linestyle='--',linewidth=0.5)
    for line in lines_coord :
        plt.axhline(y=line,color='gray',linestyle='--',linewidth=0.5)
    plt.show()
    plt.tight_layout()
    plt.savefig(plot_title)
```

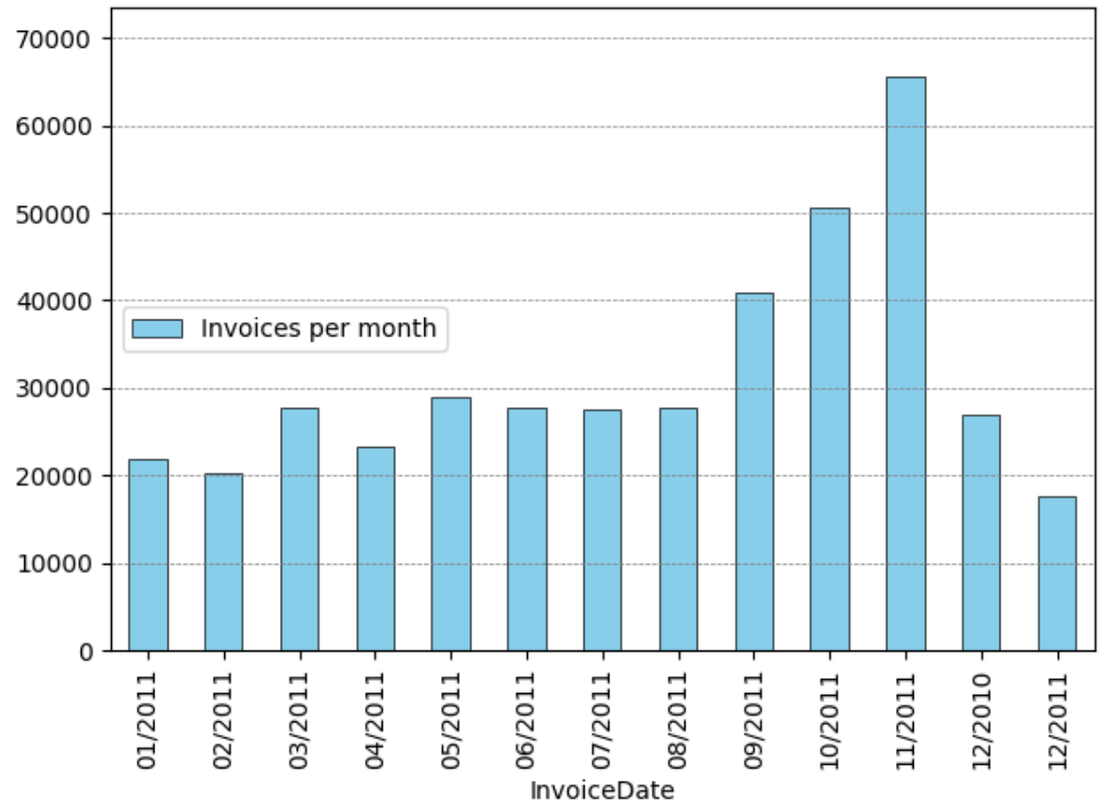
Example of function for chart plots

Exploratory data analysis: some results



- Most customers are from UK (above 30k units), next Germany, France, EIRE

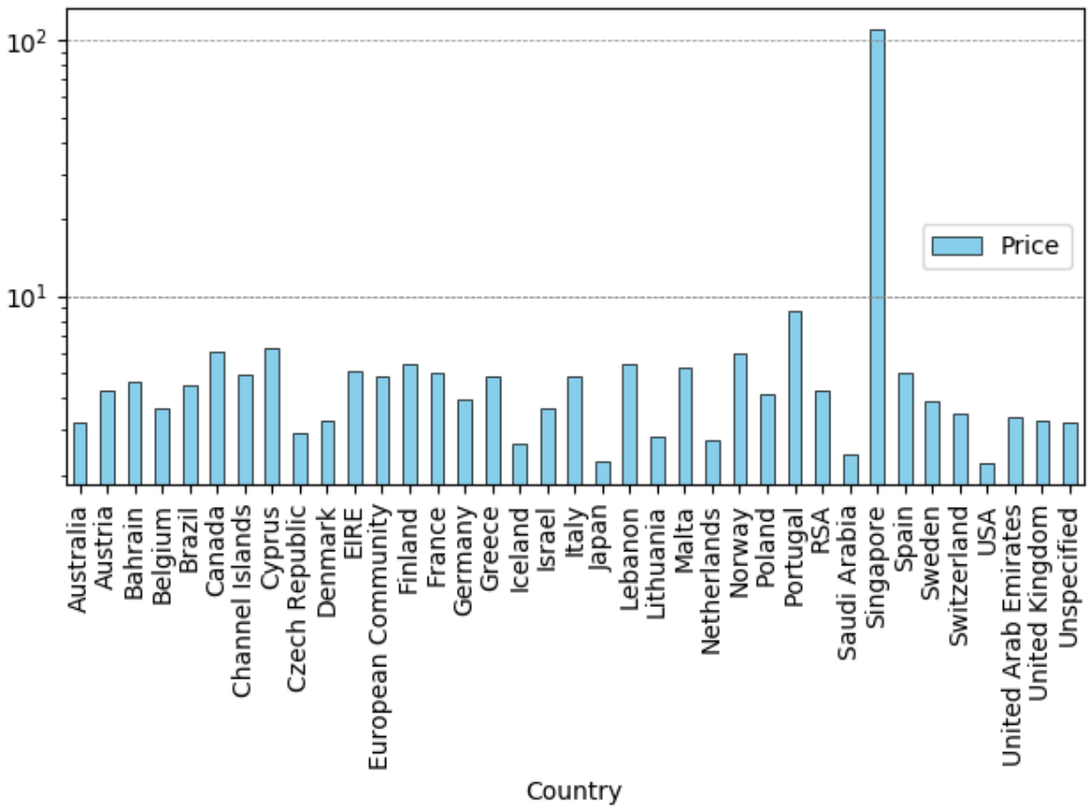
- The nationality of about 300 customers is either unspecified or accounted in “European Community”



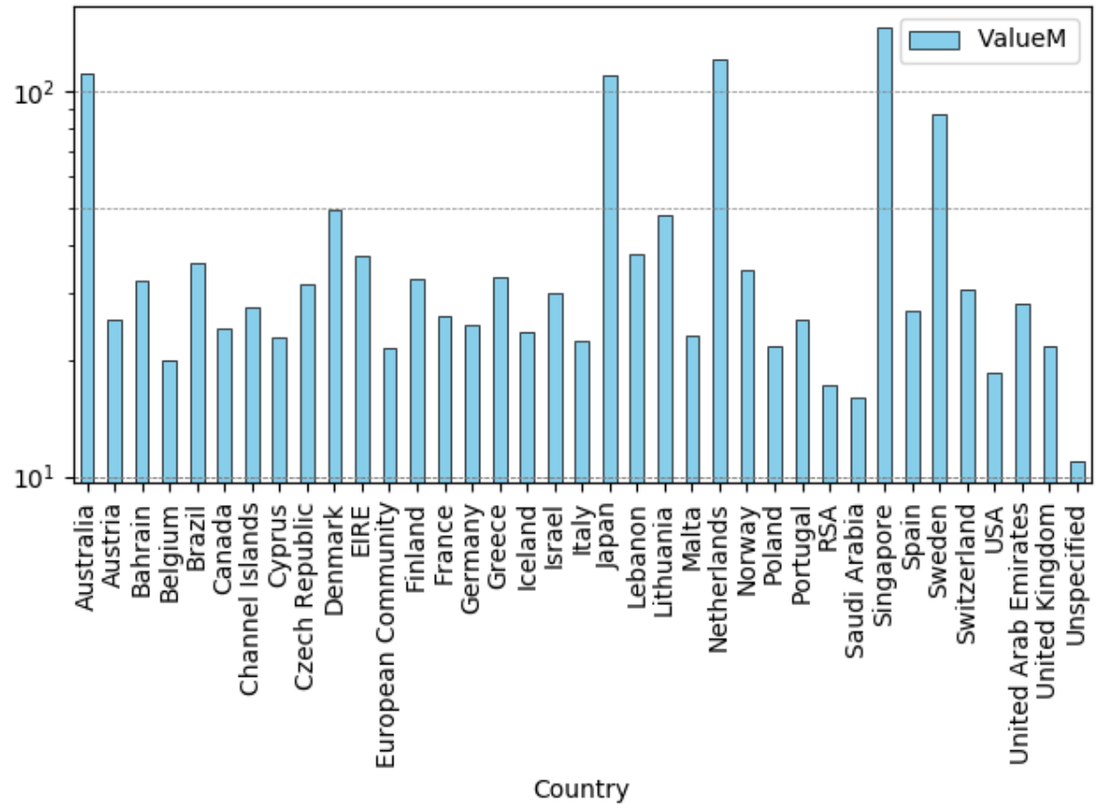
- Month with most orders is November (likely related to Christmas gifts purchase)

- Fall is the most active season of the year

Exploratory data analysis: some results

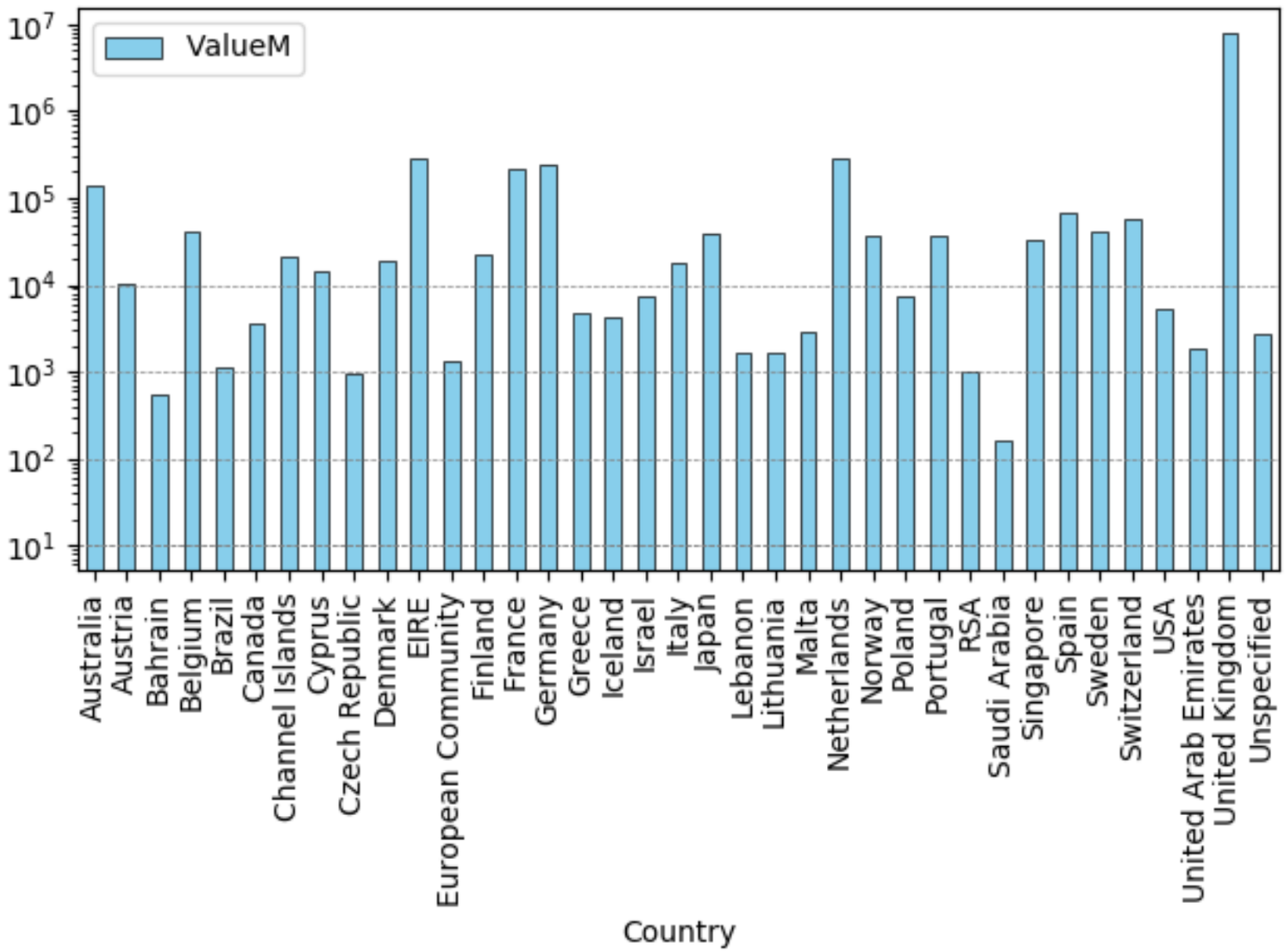


- Average item Price in British pounds for every entry
- Most customers are interested in goods with average cost below 10£
- Customers from Singapore seem to be more interested in more expensive goods



- Average items price (price times quantity) per country
- Shows that Australia, Japan, Netherlands tend to by many goods each having a small price

Exploratory data analysis: some results

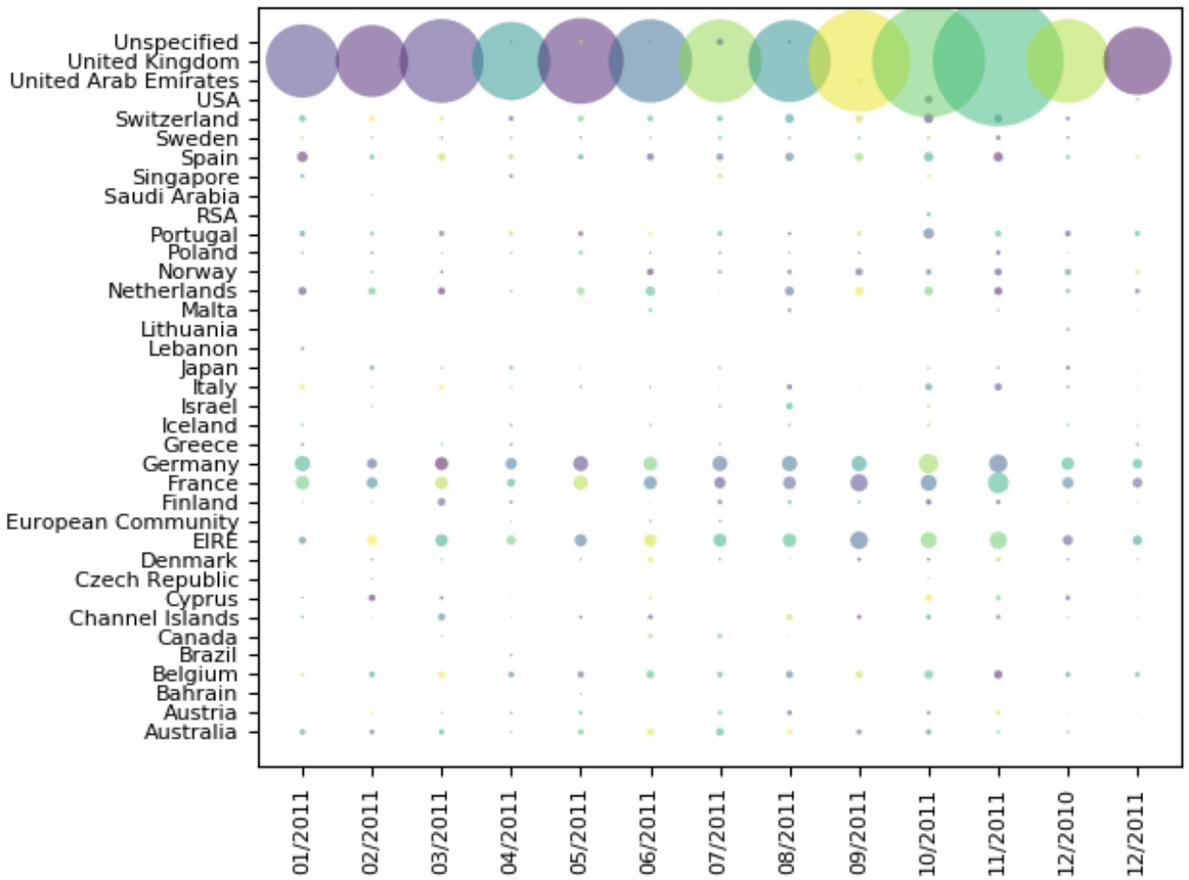


- Total revenue (British Pounds) from customers split by country

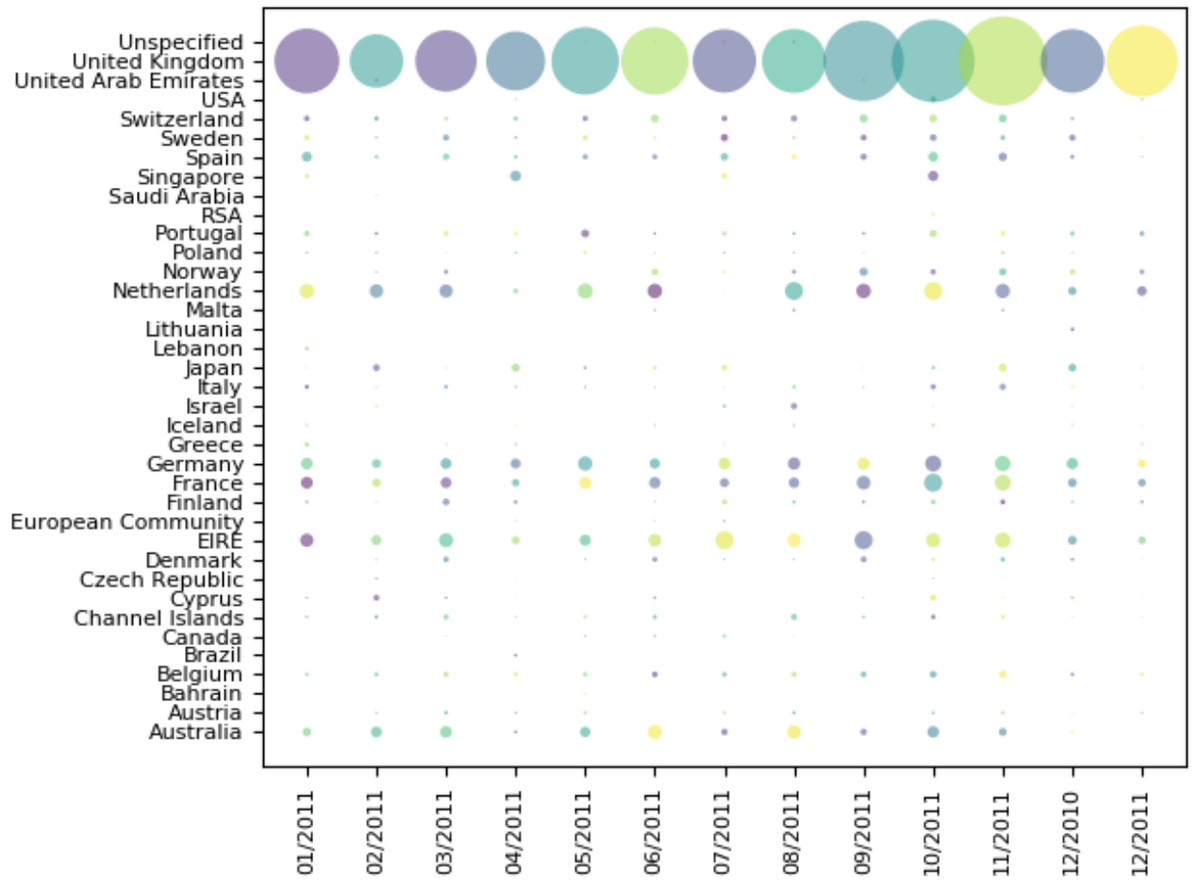
- Due to the large numbers of customers, the country brining more revenue to the company is UK

Exploratory data analysis: some results

Numbers of customers per country vs time (arbitrary normalisation)



Revenue per country vs time (arbitrary normalisation)



What interests customers the most?

- First relevant top ranked keywords (with counting): SET (571), BAG (421), HEART (411), CHRISTMAS (205), RED (152), VINTAGE (125), RETROSPOT (120), REGENCY (101), CAKE (74), SIGN (73), METAL (68), FELTCRAFT (66), DOORMAT (65), WHITE (63)

- Highlights the following
  - Customers are interested in goods sets, bags, and items for Christmas
  - In terms of design stand out goods with hearts, retro/vintage and crafted goods
  - Most popular colours of sold goods are red and white

Churned customers

- Based on 2011 list only
- Technical steps involved
  - Re-arrange the dataset (one entry per Invoice) link-to-code
  - Split the dataset based on InvoiceDate link-to-code
  - Compare the two (based on inefficient loop solution) link-to-code
- Total number of customers in first semester is 2767
- Total number of customers in second semester is 3577
- Number of churned customers is 795 (15 new customers in second semester of 2011)