**Objective:**

* To explore various APIs and examine their usefulness to data scientists through R.
* To achieve this, the capabilities of the API will be explored, the APIs’ relevance to current and potential projects as well as the packages in R that is associated with the APIs.

**Context:**

* An API is a set of protocols, tools and routines used for software building. APIs specify how components should interact with each other. In regards to this report. An API specifies the protocol used to extract data from the various data sources. Specifically how can access be gained to information available from online data bases through R.

**APIs:**

* The APIs are ranked by overall usefulness and easy of use as shown by the numbers.
* The apis, in order from most useful to least are:
  + Google maps, twitter API, flickr api, facebook API, Weather Data.

**Google Maps API:**

* Google maps is ranked first because of its general usefulness as well as relevance to current projects.
* The google Maps API is made up of the following subsequent services shown here.
* Explain roughly each API:
  + Directions: Offers route information between points. The user can specify modes of travel, request alternative routes. The general response will return the directions to get from origin to destination along with travel time, distance traveled and how to plot the path on a map.
  + Distance Matrix: provides travel time and distance from point a to point b. Multiple origin/destination pairs per request.
  + Elevation: provides elevation information for points. Takes multiple points per request.
  + Geocoding: lat/lng for addresses as well as address details (zip code, state, city etc) for requests.
  + Time zone: time zone information
  + Roads:
    - Allows the query for closest road for lat/lng points
    - Allows the points to be “snapped” to the roads and return the snapped roads
    - Speed limit information for roads is for premium account.
  + Places:
    - Places search: allows for search of locations nearby to the provided location, the search can be modified via ranking by distance, price, opening now etc.
    - Place details: using a place\_id, more detailed information regarding the place such as rating, address, phone number can be returned.
    - Place photos: allows the user to obtain google server photos for a location.
    - Place autocomplete: allows for autocomplete of a incomplete place text.
* Google maps API is ***useful to data scientists*** because it provides the processing for geographical data from a very reliable source. For example: one of the current projects in this group is to optimize ambulance car placements to reduce overall wait time. Google Maps Directions/distance matrix APIs can provide routing information for past calls, which is of essential importance. Another project involves the prediction of future delivery calls for an delivery service based in united states. Google maps geocoding takes in multiple formatted addresses and typically returns an standardized response with more details. This can be used to standardize the raw data set, making it easier to analyze.
* In the future, this service will typically provide extra information for any geographical analysis.
* The documentation for google maps API is clear and provides many examples. Therefore the capabilities of the google maps API is not tested in detail.
* Instead more effort was spent on the visualization.

**Leaflet:**

* Leaflet is a JavaScript library which uses open source map rendering data to produce interactive and custom maps.
* As a basic testing of the provided features through The R interface. 3 interactive maps are produced.
  + The first map shows the route from this building to the apartment where I live with points of interest around the map.
  + The aim of this is to visualize some of the data that google maps api (places and directions) returns.
    - The first step is to send an request for the directions between the addresses of the start and end points. The lat/lng, travel time, travel distance and an overview polyline is extracted.
    - Now just a quick note about a polyline because it comes up a lot when working with google maps direcitons API. It is basically the lat/long pairs required to plot the route information converted into bytes and then into ascii characters to form a string.It can be reversed to get the lat/long points and plotted.
    - Then you add the layers for the maps. Think of leaflet like a painting. You first paint the background, which it the map layer. Then you add on features by painting over the canvas multiple times. In the context of leaflet, through a pipeline fashion. As shown. (*run each layer added and show the output*)
      * HAVE THE MAP BE READY TO COPY AND SHOWN
      * Have the code run enough to illustrate each map.
    - For the points of interest, I followed the same procedure. Make the request, parse the data for name, at/long of points of interest as well as the icon and add it as layers onto the map.
    - Explain crime data roughly, percentage proportional to radius of circle markers. Demonstrate zoom in, zoom out, click on label.
    - Stephane wants to see capability, explain what the project is about, mention weakness of overview polyline.
    - Why is leaflet useful:
      * Visualization of geographical data
      * Allows interactive contents through zooming, labels that adds to presentations, or in general to be able to express more data.
    - SUMS up google

**Twitter:**

* Twitter was ranked 2nd in terms of usefulness to data scientists because of the massive amounts of data available as well as convient R packages allowing access to the API. Tweets and user information is typically publicly available, and this large amount of data can lead to interesting things.
* Search API was first explored:
  + Downfall: 1 week time limit, historical data access can be paid for.
  + Upside: powerful query language, allows querying for filters for media, attitudes, hashtags.
  + A search for tweets containing #NRC,#NRC-Canada and text of tweet to conduct NRC was contacted (roughly 100 tweets was gathered). A word clout is produced.
  + This is just one the cool things you can do. Of course with massive amounts of text that is easily categorized based on user information, hashtags, etc. Twitter is an excellent source of data for **Data scientists.**
* Another interesting Feature is the streaming API. Which allows public access to newly produced tweets. An sample stream, of tweets mentioning CNN is collected. The streaming connection is handled by a simple R package.
  + As you can see a single tweet contains plenty of meta data which can be helpful for any sentiment analysis and categorizing tweets in general.
  + An potential is to stream for data when interesting events that could be useful comes up. For example: stream for tweets mentioning Donald Trump, store in a data base and for the future conduct a research on attitudes regarding Donald Trump.
* One last possibility is to contact social network alnayses on followers of users and who the user follows because this data is generally available for public users.
* As a summary, Massive of data is available through Twitter API, thus makes it very useful.

**Flickr:**

* Flickr is ranked 3rd in terms of usefulness because of the details it provide. Flickr is an photo hosting/sharing website. This means the metadata of photos, which are typically stripped away on the other APIs discussed here is aviable.
* A mini-project was to search for tweets and photos about Canada day 2015, restricting the search to be around the Canadian parliament.
* Unfortunantly, due to the time restrictions of the twitter search API, this was not possible
* For flickr, some of the results are shown here. As you can see, the search proved quite effective in terms of relevance and restrictions on locations.
* The metadata sample is shown here:
  + You can spot camera model, orientiaion, details regarding pictures taken etc.
* It is because of this detail that flickr is ranked 3rd.

**Facebook:**

* Facebook is ranked 4th due to major flaw that user information is not publicly available. However the amount of data available is impressive. Facebook API offers the ability to access public pages/groups as well as the posts, comments and replies to comments with the search API.
* Two mini-projects were tried. The first is a search for pages that mention flowers. This historgram shows the distribution of the responses marked by country. Around 500 pages were collected, a vast majority did not supply a location information. The same with the second project where 500 posts from a travelling page were parsed for location tag information, only a minority had. For interest, the most popular place visited that is tagged is india.

**Weather Data:**

Weather Web Scraping:

* Data Collection:
  + The goal is to collect the current weather condition and the forecasted weather from the environment Canada website.
    - These two pictures hows the interface.

The main packages used are rvest and xml2 for parsing the source code.

* + A css tag was used to extract the information needed by manually finding it from the source code of the website.
    - <http://www.computerhope.com/issues/ch000746.htm>
  + The website is updated every hour. So in order to collect the data, a windows R script was run to append information to the excel file.
    - The package used is taskscheduleR, which also provides an R studio plugin to run the script.
    - This package interfaces with the taskScheduler of windows through schtasks.ext run in command line.
  + The problem was that if the power source is unplugged, the script would not run. In order to address it, the xml file containing the parameters which specify how the task is run needs to be modified. This is again done with the XML package, then the same schtask tool is called to run the modified version of the file.
  + Now to get the predications to align with current time, columns are lagged. (explain process)Then the other columns are calculated.
* Data Processing: explain what each graph is:
  + Mention error bars,
  + Signed errors