# Dublin R
# El Dorado competition 2016

**Mihai Todor, Software Engineer,**
**Kevin Eid, Data Engineer,**
*team miner49er*

# Introduction

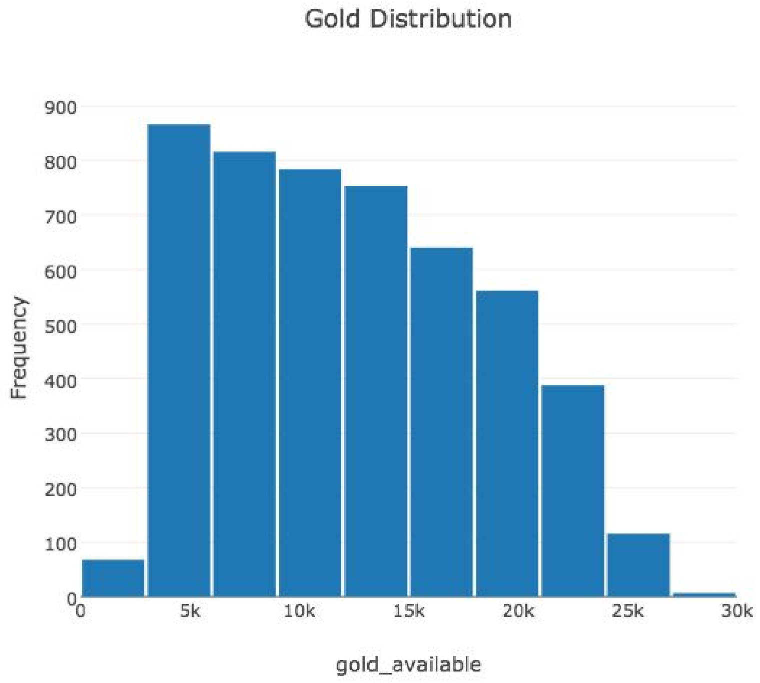
1. **Used Python**
    - Scipy, Scikit-Learn, Matplotlib / Plotly

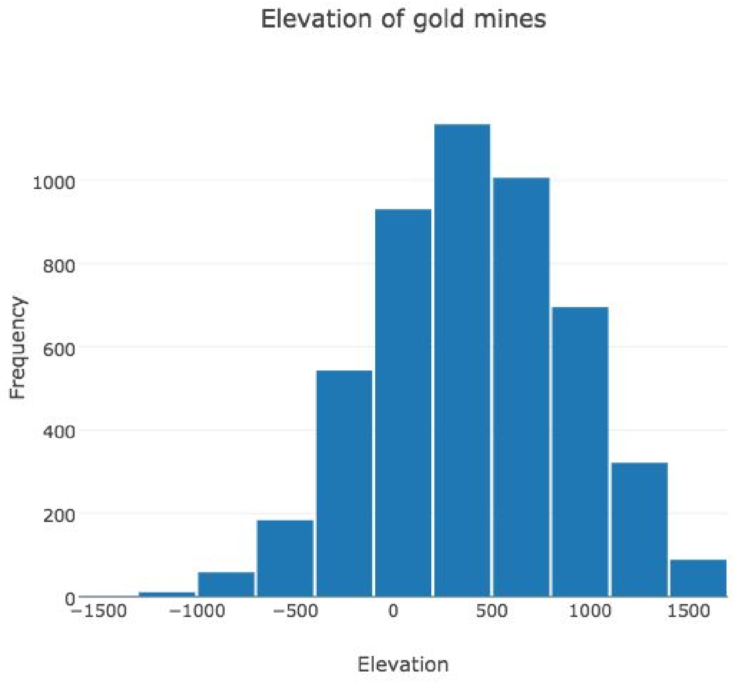
2. **Strategy**
    - Get the optimal cluster in terms of amount of gold and elevation
    - Find which parcels belong to this cluster
    - Pick the 5 most profitable parcels and focus our investment plan on them

# Data Preprocessing

1. Elevation: continuous → categorical values:
    - 3 elevation categories for fixed costs
    - 3 elevation categories for variable costs
2. Linear Regression on the costs_data

# Data Exploration



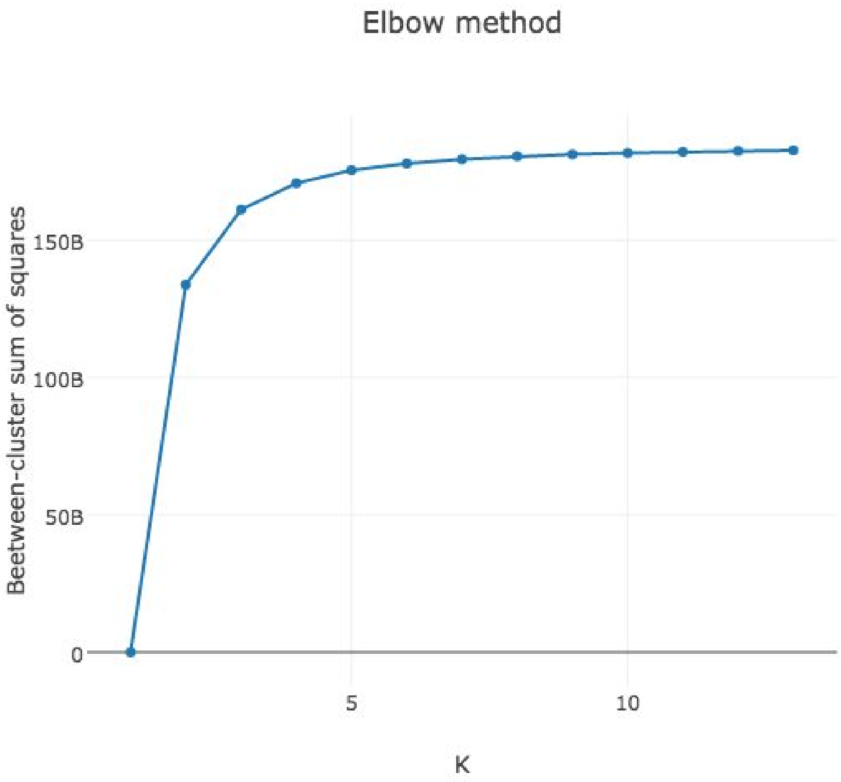


# Clustering - K-means

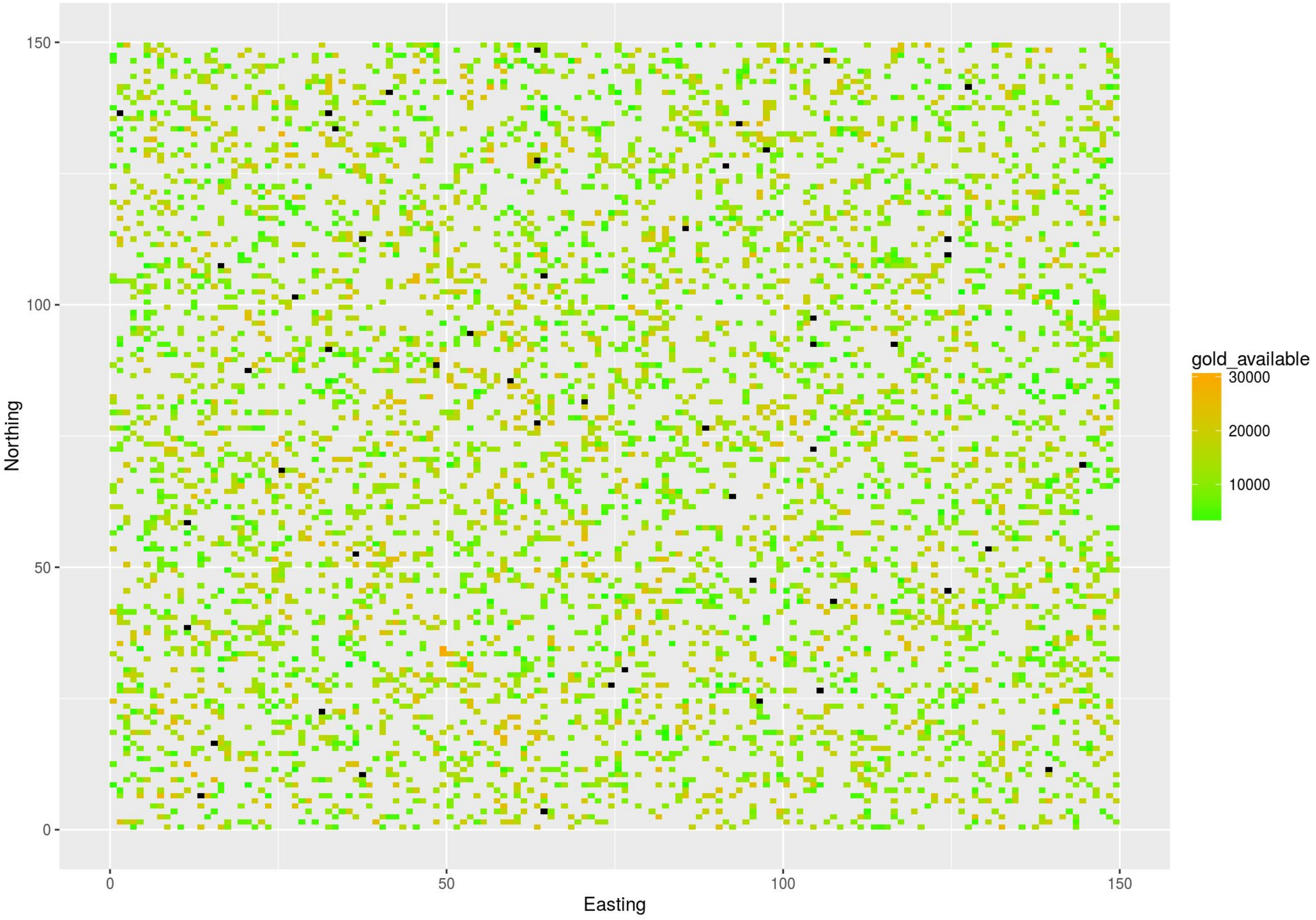
- Clustering on **gold_available** and **elevation**

- **Elbow method**: computing and plotting the **between-cluster sum of squares.**
  - bss = total_sum_of_squares - within-cluster_sum_of_squares

- K = 3

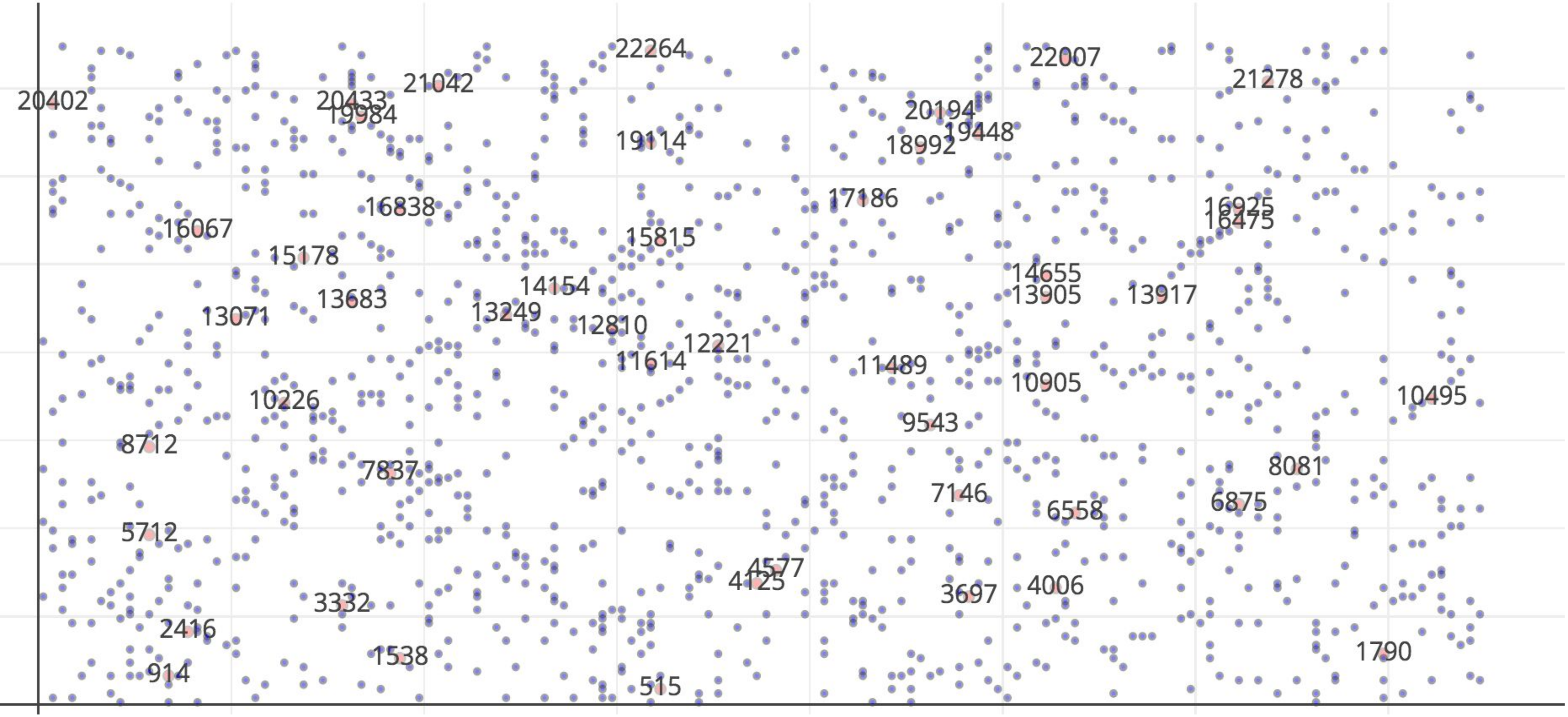


Plot of the Sample and Auction Data
(Auction Parcels coloured Black)



# First filtering of the parcels

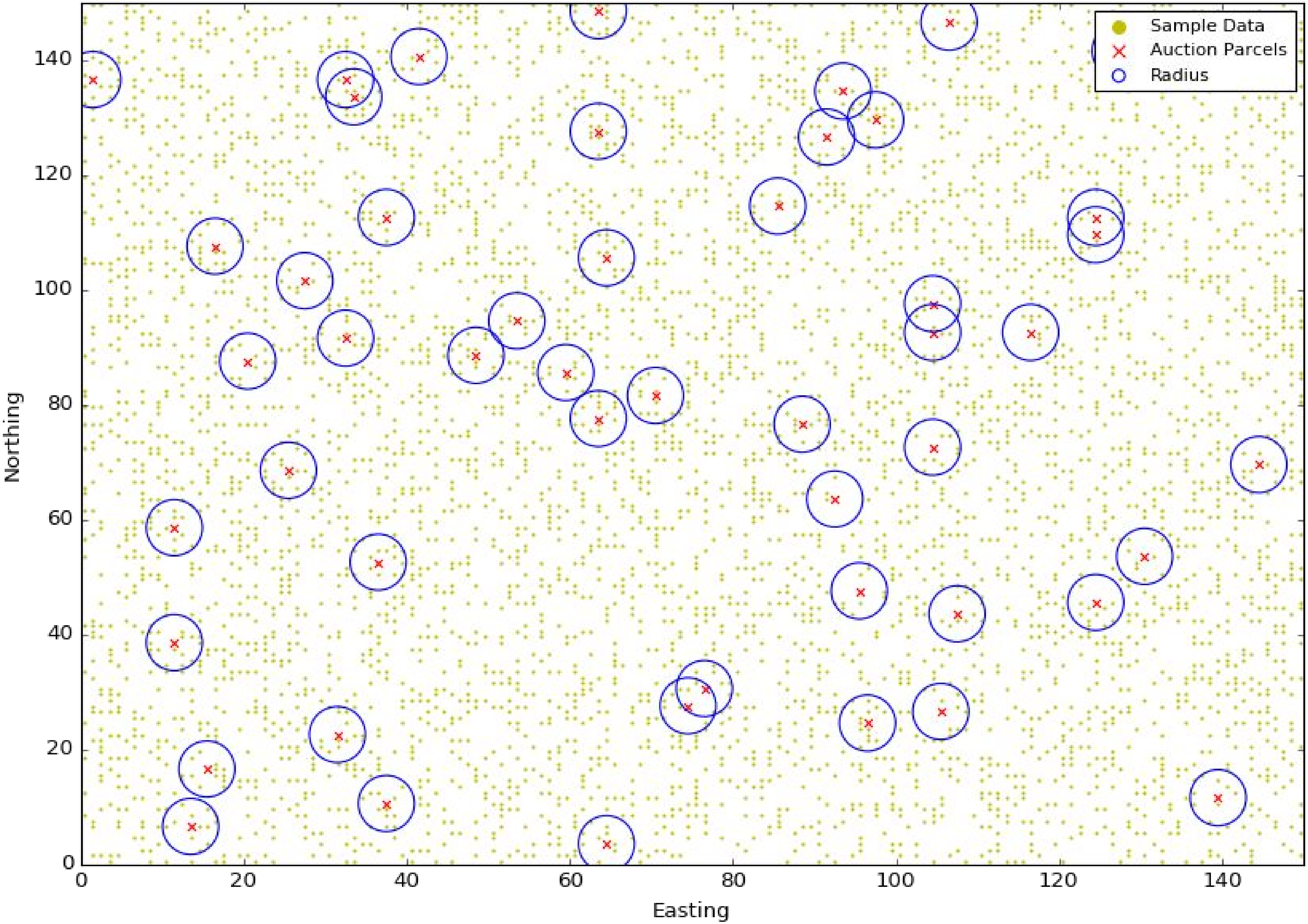
```
top_parcel_ids = [10226, 7837, 19114, 20194,
11489,10905,1790,13249,14154,12810,1614,12221]
```



# Another idea for filtering

- Compute average gold amount around each parcel

- Hope that Mick did not choose the gold quantities under neighbouring parcels at random :)


Sample Data
Auction Parcels
Radius
Northing
Easting
0
20
40
60
80
100
120
140
0
20
40
60
80
100
120
140

# Empirical area radius selection

- Try radius = 1 .. 10
- Sort parcels by the estimated profit
- Eyeball the smallest radius for which the parcel order doesn’t change (much). We chose radius = 3

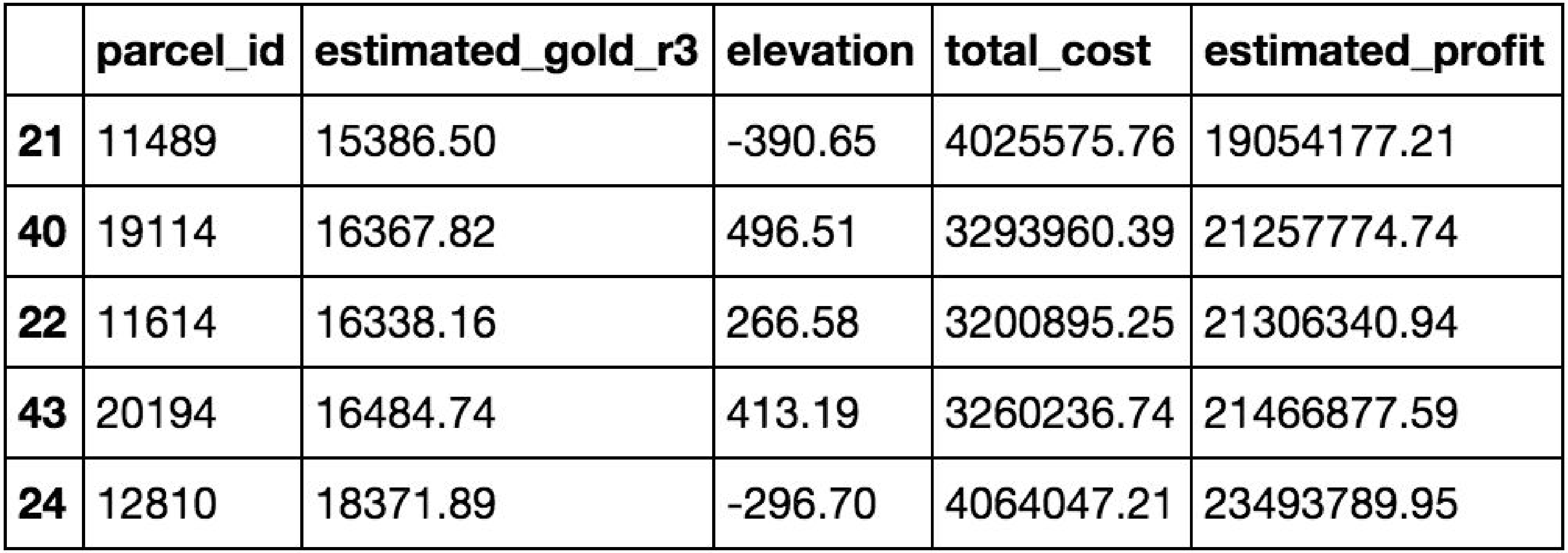
# Computing estimated extraction costs and profits

- Select the top 10 parcels which would yield the biggest profit for the selected radius

# Match best parcels yielded by both methods

- Intersect the selected parcels with the ones produced by clustering

- Select the top 5 most promising parcels

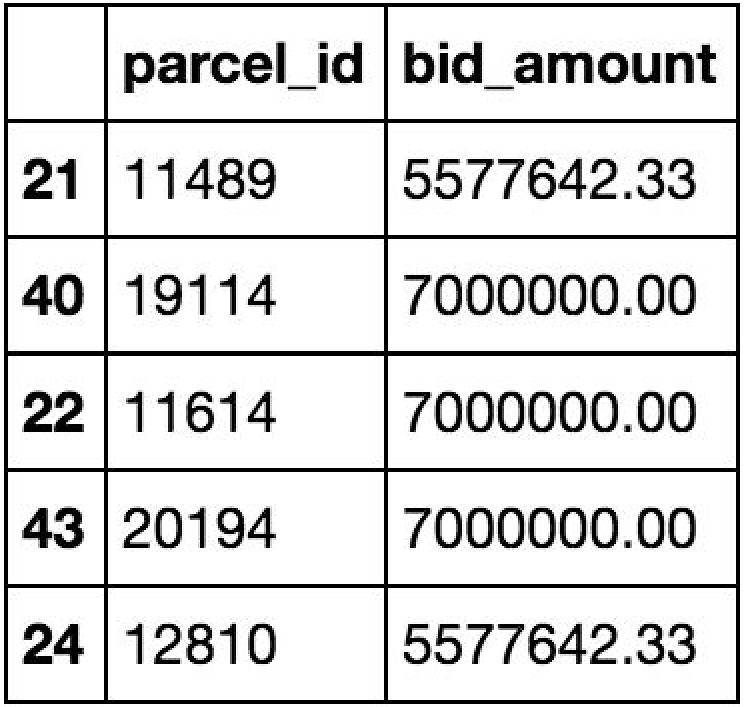


| | parcel_id | estimated_gold_r3 | elevation | total_cost | estimated_profit |
|---|---|---|---|---|---|
| **21** | 11489 | 15386.50 | -390.65 | 4025575.76 | 19054177.21 |
| **40** | 19114 | 16367.82 | 496.51 | 3293960.39 | 21257774.74 |
| **22** | 11614 | 16338.16 | 266.58 | 3200895.25 | 21306340.94 |
| **43** | 20194 | 16484.74 | 413.19 | 3260236.74 | 21466877.59 |
| **24** | 12810 | 18371.89 | -296.70 | 4064047.21 | 23493789.95 |

# Bidding strategy using empirical Gaussian distribution

- Remove the gold extraction costs for the selected 5 parcels from the total budget

- Bid more for the middle 3 parcels



| | parcel_id | bid_amount |
|---|---|---|
| **21** | 11489 | 5577642.33 |
| **40** | 19114 | 7000000.00 |
| **22** | 11614 | 7000000.00 |
| **43** | 20194 | 7000000.00 |
| **24** | 12810 | 5577642.33 |

# El Dorado 2.0

- Run the second filtering on the parcels from the optimal cluster instead of all the parcels
- Try a different bidding strategy
- Using classifiers to predict to which cluster a parcel belongs to
- Verify results :)

# Code

https://github.com/mihaitodor/el_dorado

# THANK YOU