Πανεπιστήμιο Μακεδονίας
Σχολή Διοίκησης Επιχειρήσεων
ΠΜΣ Αναλυτική των επιχειρήσεων και Επιστήμη των δεδομένων
Εξόρυξη Δεδομένων και Προχωρημένες Τεχνικές Προβλεπτικής Αναλυτικής

# Μελέτη περίπτωσης AirBNB listings (Αθήνα)

---

Εφαρμογή μετασχηματισμού, οπτικοποίησης και μοντέλων παλινδρόμησης για την πρόβλεψη της τιμής διαμονής

Ομάδα εργασίας:
Αθανασιάδου Σωτηρία - BAD21003 - bad21003@uom.edu.gr
Δεληγιάννης Δημήτρης - BAD21007 - bad21007@uom.edu.gr
Λαζαρίδου Βασιλική - BAD21020 - bad21020@uom.edu.gr
Μητσιάνης Χρήστος - BAD21022 - bad21022@uom.edu.gr

Διδάσκων: Δρ. Άγγελος Μάρκου, Αναπληρωτής Καθηγητής

Ακαδημαϊκό έτος 2021 - 2022

Θεσσαλονίκη, Φεβρουάριος 2022

Η σελίδα είναι σκοπίμως λευκή

# Πίνακας Περιεχομένων

# Εισαγωγή

Με την βοήθεια της ιστοσελίδας Inside Airbnb (http://insideairbnb.com/get-the-data.html), η οποία συλλέγει από την ιστοσελίδα της Airbnb δημόσια διαθέσιμες πληροφορίες σχετικά με τις καταχωρήσεις καταλυμάτων (listings) της Airbnb σε διαφορετικές πόλεις, πραγματοποιήθηκε η συγγραφή της παρούσας εργασίας η οποία πραγματεύεται την πρόβλεψη της τιμής καταλύματος (price σε $) από ένα σύνολο καταλυμάτων της πόλης της Αθήνας, εφαρμόζοντας κατάλληλα προβλεπτικά μοντέλα και αφού πρώτα διενεργήθηκε κατάλληλη προεπεξεργασία στο σύνολο των δεδομένων στο αρχείο listings.csv.

Τα δεδομένα που συλλέχθηκαν από το αρχείο http://data.insideairbnb.com/greece/attica/athens/2021-10-25/data/listings.csv.gz αφορούν δεδομένα για τα καταλύματα της πόλης της Αθήνας που ήταν διαθέσιμα στην πλατφόρμα στις 25/10/2021.

Επιμέρους ζητούμενα της εργασίας είναι:

α) η ελαχιστοποίηση του δείκτη Root Mean Squared Error – RMSE στο test set και

β) ο προσδιορισμός των πιο σημαντικών μεταβλητών που επηρεάζουν την τιμή του καταλύματος και η σχέση αυτών των μεταβλητών με την εξαρτημένη μεταβλητή.

# Εισαγωγή & Προεπεξεργασία δεδομένων

Για την παρούσα μελέτη περίπτωσης έγινε λήψη των δεδομένων που αφορούν εγγραφές - καταχωρήσεις στην πλατφόρμα Airbnb τον Οκτώβριο του 2021.

Το σύνολο των αρχικών εγγραφών ήταν:

Το σύνολο των αρχικών μεταβλητών ήταν:

Για την πρώτη φάση της προεπεξεργασίας και επιλογής των δεδομένων χρησιμοποιήθηκαν οι βιβλιοθήκες της R, dplyr και tidyverse

- Έγινε αφαίρεση των μη διαθέσιμων καταχωρήσεων.

```
airbnb <- filter(airbnb, has_availability=='t')
```

- Πραγματοποιήθηκε αφαίρεση των καταχωρήσεων που έχουν NA ή 0 reviews, η παρατήρηση έδειξε πως όσα έχουν NA ή 0 έχουν γενικότερα ελλιπή δεδομένα και στις υπόλοιπες μεταβλητές.

```
airbnb <- filter(airbnb, number_of_reviews > 0)
```

- Στην συνέχεια αφαιρέθηκαν οι εξής μεταβλητές για λόγους μη σχέσης τους με την πρόβλεψη της τιμής, ελλειπουσών τιμών και υψηλής συσχέτισης με άλλες μεταβλητές που παρέμειναν στο σύνολο των δεδομένων.

```
airbnb <- select(airbnb, -c(
  id,
  listing_url,
  scrape_id,
  name,
  description,
  picture_url,
  calendar_last_scraped,
  calendar_updated,
  has_availability,

  host_url,
  host_name,
  host_since,
  host_id,
  host_location,
  host_about,
  host_response_rate,
  host_acceptance_rate,
  host_thumbnail_url,
  host_picture_url,
  host_neighbourhood,
  host_total_listings_count,
  host_has_profile_pic,
  host_verifications,
```

```
  neighbourhood,
  neighbourhood_group_cleansed,
  neighborhood_overview,
  bathrooms,
  # remove bedrooms because of NAs
  bedrooms,
  # remove scores_rating, same as scores_value
  review_scores_rating,

  minimum_nights,
  minimum_minimum_nights,
  minimum_maximum_nights,
  minimum_nights_avg_ntm,
  maximum_nights,
  maximum_minimum_nights,
  maximum_maximum_nights,
  maximum_nights_avg_ntm,
  calculated_host_listings_count,
  calculated_host_listings_count_entire_homes,
  calculated_host_listings_count_private_rooms,
  calculated_host_listings_count_shared_rooms,

  license))
```

- Ακολούθως έγινε μετατροπή των χαρακτήρων t και f (true - false) με 1 και 0 αντίστοιχα στις παρακάτω μεταβλητές

```
airbnb <- mutate(airbnb,
     host_is_superhost = as.integer(ifelse(host_is_superhost == 't',
1, 0)),
                                        host_identity_verified       =
as.integer(ifelse(host_identity_verified == 't', 1, 0)),
     instant_bookable = as.integer(ifelse(instant_bookable == 't', 1,
0)))
```

- Κάνοντας χρήση της βιβλιοθήκης stringr έγινε αφαίρεση του συμβόλου $ από την μεταβλητή price

```
airbnb$price<-as.integer(str_replace_all(airbnb$price,"\\$|,",""))
```

- Με την χρήση της βιβλιοθήκης lubridate οι ακόλουθες μεταβλητές μετατράπηκαν σε μεταβλητές τύπου date

```
airbnb <- mutate(airbnb,
                 first_review = ymd(first_review),
                 last_review = ymd(last_review),
                 last_scraped = ymd(last_scraped)
                 )
```

- Δημιουργήθηκαν οι νέες μεταβλητές: listing_dur η οποία δέχεται την απόλυτη τιμή της διαφοράς των ημερών last_scraped και first_review, η lastrev_firstrev_dur η οποία δείχνει το χρονικό διάστημα μεταξύ της πρώτης και τελευταίας αξιολόγησης και η from_lastrev_dur η οποία δείχνει τις μέρες που έχουν περάσει μεταξύ της τελευταίας αξιολόγησης και της άντλησης των δεδομένων.

```
airbnb <- airbnb %>%
          mutate (
                                                    listing_dur   =
abs(as.integer(difftime(last_scraped,first_review,      units      =
'days'))),
                                                lastrev_firstrev_dur   =
abs(as.integer(difftime(last_review,first_review, units = 'days'))),
                                                   from_lastrev_dur   =
abs(as.integer(difftime(last_scraped,last_review, units = 'days'))))
```

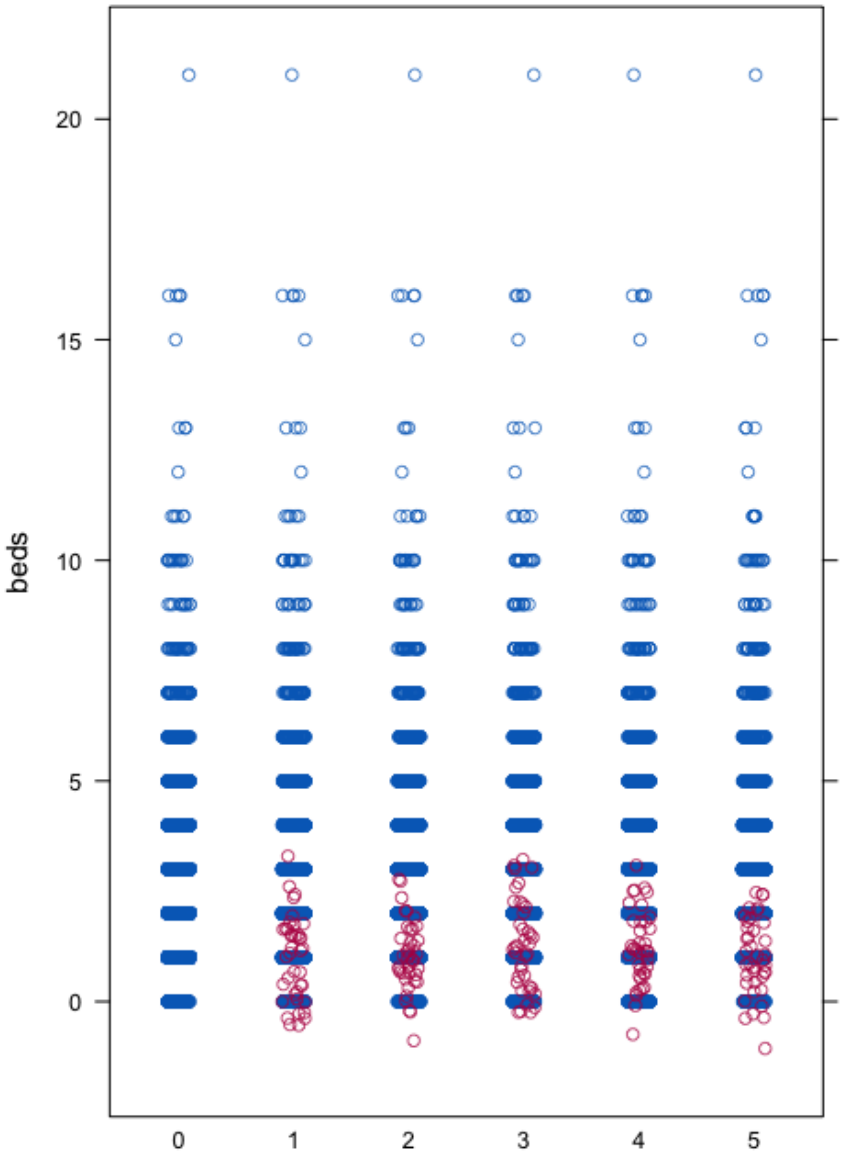
- Στην συνέχεια πραγματοποιήθηκε κειμενική επεξεργασία στην μεταβλητή bathrooms_text με αντικατάσταση του half με τον αριθμό 0.5. Στην συνέχεια δημιουργήθηκε η μεταβλητή bath_sum η οποία λαμβάνει το αριθμητικό μέρος της bathrooms_text δηλαδή καταχωρείται ο αριθμός των μπάνιων που έχει κάθε κατάλυμα. Επειδή τα μπάνια χαρακτηρίζονται ως shared και private δημιουργήθηκαν οι μεταβλητές bath_shared και bath_private οι οποίες παίρνουν τις τιμές 1 ή 0 αντίστοιχα αν βρεθούν οι χαρακτήρες “share” και “priv” στην μεταβλητή bathrooms_text. Και στην συνέχεια αφαιρέθηκαν οι εγγραφές που έχουν NA στην μεταβλητή bath_sum

```
airbnb <- airbnb %>%
          mutate (
# Replace half or Half in bathrooms_text with 0.5
                                bathrooms_text  =  gsub("half",  "0.5",
bathrooms_text, ignore.case = TRUE),

# Create new column bath_sum and extract only numeric values from
bathrooms_text
                                                      bath_sum   =
as.numeric(str_extract(bathrooms_text,"\\d\\.{0,1}\\d{0,1}")),

# 1 or 0 if bathroom/s shared or private, Drop bathrooms_text
                                                    bath_shared    =
as.integer(ifelse(str_detect(bathrooms_text,"share"), 1,0)),
                                                   bath_private   =
as.integer(ifelse(str_detect(bathrooms_text,"priv"), 1,0)),
                 bathrooms_text = NULL)%>%
```

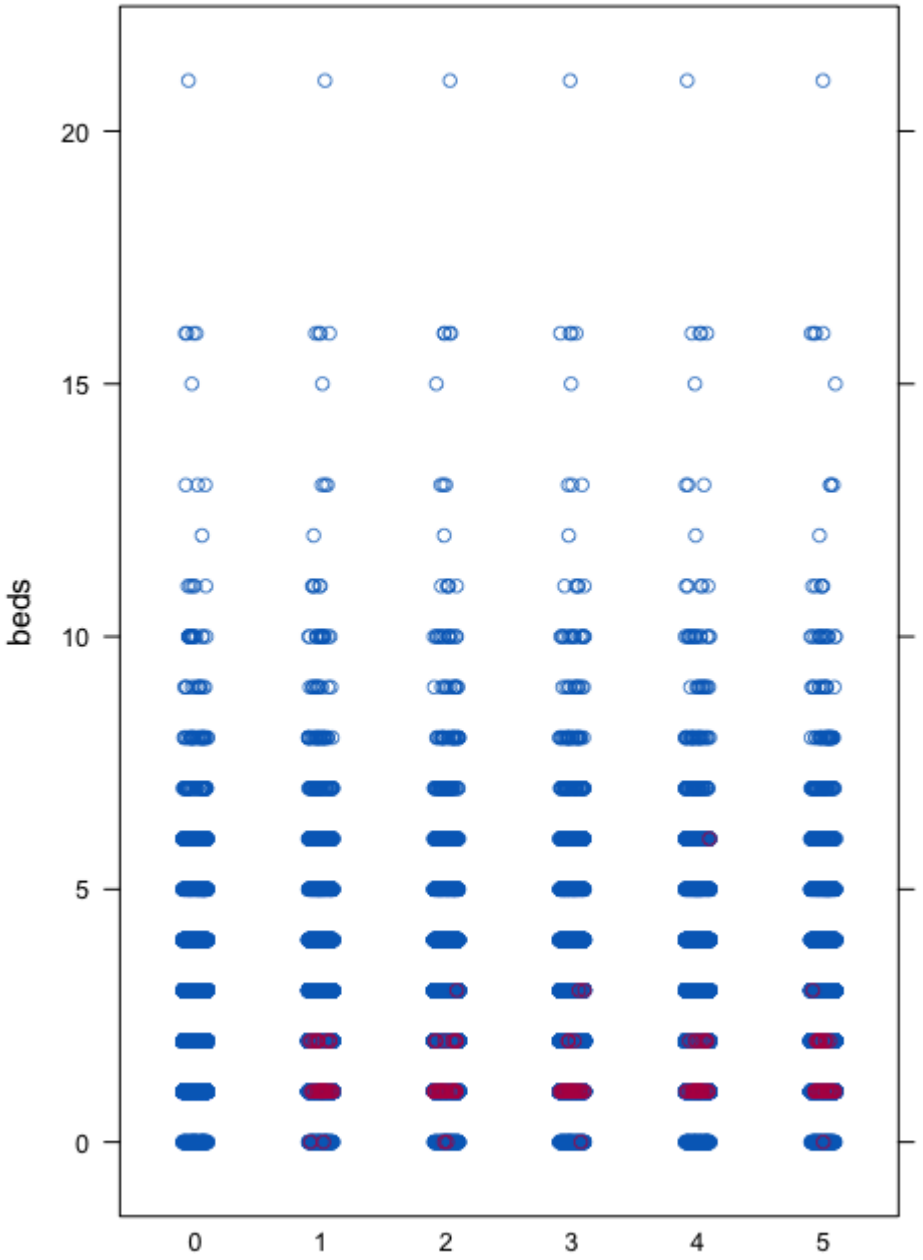
```
filter (
            !is.na(bath_sum))
```

- Αντικατάσταση ελλειπουσών τιμών. Στην συγκεκριμένη φάση της προεπεξεργασίας των δεδομένων χρησιμοποιήθηκε η βιβλιοθήκη mice: Multivariate Imputation by Chained Equations. Μετά από δοκιμές διαφόρων αλγορίθμων συμπλήρωσης τιμών ξεχώρισε ο αλγόριθμος PMM ή predictive mean matching. Ο αλγόριθμος αναλύει όλα τα δεδομένα και ανάλογα με τις τιμές στις υπόλοιπες μεταβλητές εισάγει στην εγγραφή μια μέση τιμή στην ελλείπουσα τιμή ανάλογα με τις συγγενικές της εγγραφής. Στην συνέχεια έγιναν διαγράμματα του τρόπου με τον οποίον συμπληρώνονται - τοποθετούνται οι ελλείπουσες τιμές. Και έγινε επιλογή και αντικατάσταση τους στο βασικό σύνολο δεδομένων.

```
#NAs handling
library(mice)
library(glmnet)
# mice: Multivariate Imputation by Chained Equations
impute <- mice(airbnb,
               m=5,
               seed=123,
               method =
                    # Experiments with different methods (final=pmm,
3rd iteration)
                 #"norm.predict"
                 #"lasso.select.norm"
                 "pmm"
                 #"midastouch"
               )
#Print impute object and create plots to examine imputation
print(impute)
stripplot(impute, beds)
stripplot(impute, host_listings_count)
stripplot(impute, review_scores_value)

# Complete data imputation and remove "impute" object
airbnb <- complete(impute, 3)
rm(impute)
```




Διάγραμμα συμπλήρωσης τιμών με τον αλγόριθμο lasso.select.norm, στον οριζόντιο άξονα βρίσκονται οι επαναλήψεις. Με κόκκινο χρώμα συμβολίζονται οι τιμές που συμπληρώνονται. Διακρίνονται οι τιμές να λαμβάνουν εσφαλμένες τιμές και κάτω του μηδενός.




Διάγραμμα συμπλήρωσης τιμών με τον αλγόριθμο pmm, στον οριζόντιο άξονα βρίσκονται οι επαναλήψεις. Με κόκκινο χρώμα συμβολίζονται οι τιμές που συμπληρώνονται. Διακρίνονται οι τιμές να λαμβάνουν ορθές τοποθετήσεις σε σχέση με τις υπόλοιπες. Επιλέχθηκε η 3η επαναληψη.

- Στην συνέχεια δημιουργήθηκαν οι μεταβλητές total_amenites όπου μετράει το πλήθος των παροχών κάθε καταλύματος και η is_top_100 η οποία λαμβάνει τις τιμές 1 ή 0 αν είναι στις κορυφαίες 100 στις κριτικές.

```
airbnb <- airbnb %>% mutate(
# Amenities count
total_amenities = ifelse(nchar(amenities)>2, str_count(amenities, ',') +1, 0),
```

```
    # Is in top 100 in ranking
    is_top_100    =    ifelse(rank(-number_of_reviews,    ties.method    =
"average") <= 100, 1, 0))
```

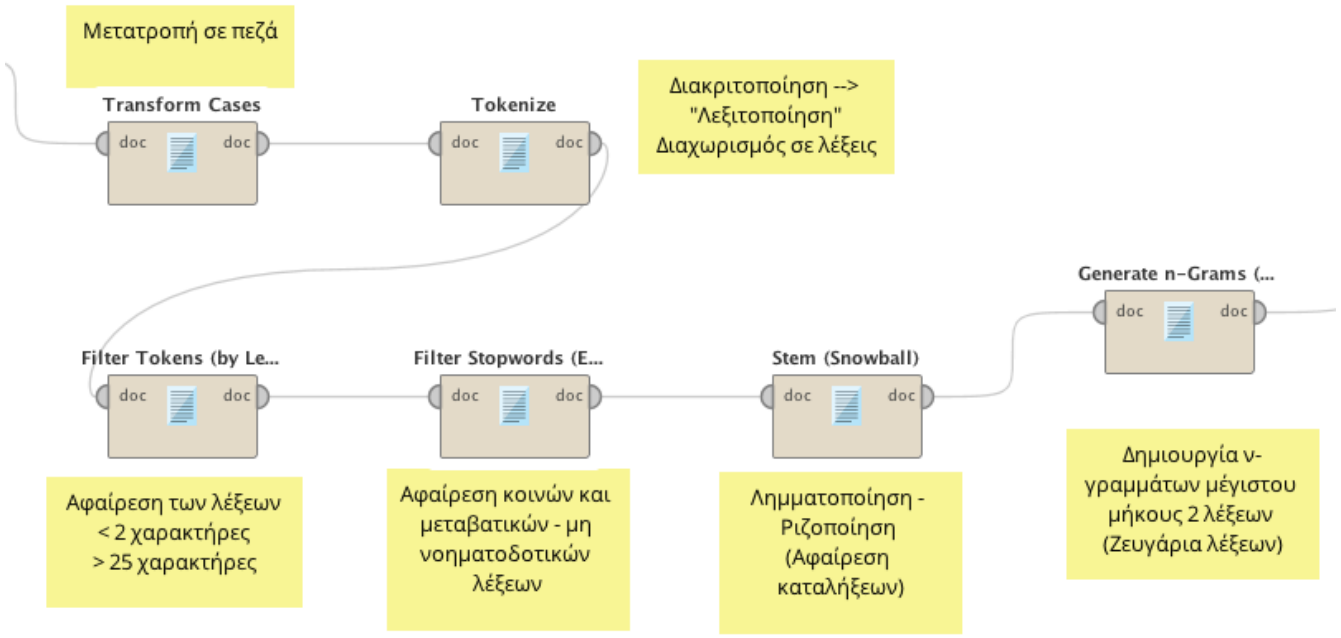
- Πραγματοποήθηκε αφαίρεση των κενών χαρακτήρων από τις κατηγορικές μεταβλητές.
- Με βάση τις κατηγορικές μεταβλητές δημιουργήθηκαν σετ με ψευδο-μεταβλητές 0/1

```
    # Creating dummy variables 0/1 on neighbourhood_cleansed, room_type,
    property_type, host_response_time
    library(dummies)
    nb_dummy <- dummy(airbnb$neighbourhood_cleansed, sep = "_")
    room_type_dummy <- dummy(airbnb$room_type, sep = "_")
    property_type_dummy <- dummy(airbnb$property_type, sep = "_")
    host_response_time_dummy  <-  dummy(airbnb$host_response_time,  sep  =
"_")
```

- Ενοποιήθηκαν τα δεδομένα σε ένα σύνολο δεδομένων

```
airbnb <- airbnb %>%
  cbind(nb_dummy) %>%
  cbind(room_type_dummy) %>%
  cbind(property_type_dummy) %>%
  cbind(host_response_time_dummy)
```

- Στην συνέχεια πραγματοποιήθηκε κειμενική ανάλυση στο λογισμικό Rapidminer. Η κειμενική ανάλυση βασίζεται στον αλγόριθμο TF-IDF όπου εξετάζει την συχνότητα εμφάνισης ενός λήμματος σε ένα κείμενο σε σχέση με την συχνότητα εμφάνισης του λήμματος στο σύνολο των κειμένων. Συνοπτικά πραγματοποιήθηκε μετατροπή σε πεζά, λημματοποίηση, φιλτράρισμα μη νοηματοδοτικών λέξεων, ριζοποίηση (streming) και δημιουργία n-γραμμάτων με n=2. Τέλος έγινε αφαίρεση του 20% των κορυφαίων σε συχνότητα λημμάτων και του 10% ποιο σπάνιων




Η διαδικασία επεξεργασίας στο Rapidminer.

- Στη συνέχεια οι βασικές μεταβλητές του συνόλου δεδομένων πήραν τα αρχικά bs., οι dummy μεταβλητές τα αρχικά dm, και οι μεταβλητές της κειμενικής ανάλυσης τα αρχικά tf.

```
# Renaming columns names in 3 categories bs = basic , dm = dummy, tf = tfidf
colnames(airbnb_2)[1:35] <- paste('bs', colnames(airbnb_2[1:35]), sep = ".")
colnames(airbnb_2)[36:131] <- paste('dm', colnames(airbnb_2[36:131]), sep = ".")
colnames(airbnb_2)[132:314]<- paste('tf', colnames(airbnb_2[132:314]), sep = ".")
```

- Έγινε μετατροπή των μεταβλητών τύπου χαρακτήρων σε κατηγορικές μεταβλητές

```
airbnb_2 <- as.data.frame(unclass(airbnb_2), stringsAsFactors = TRUE)
```

- Έγινε λογαριθμοποίηση της τιμής

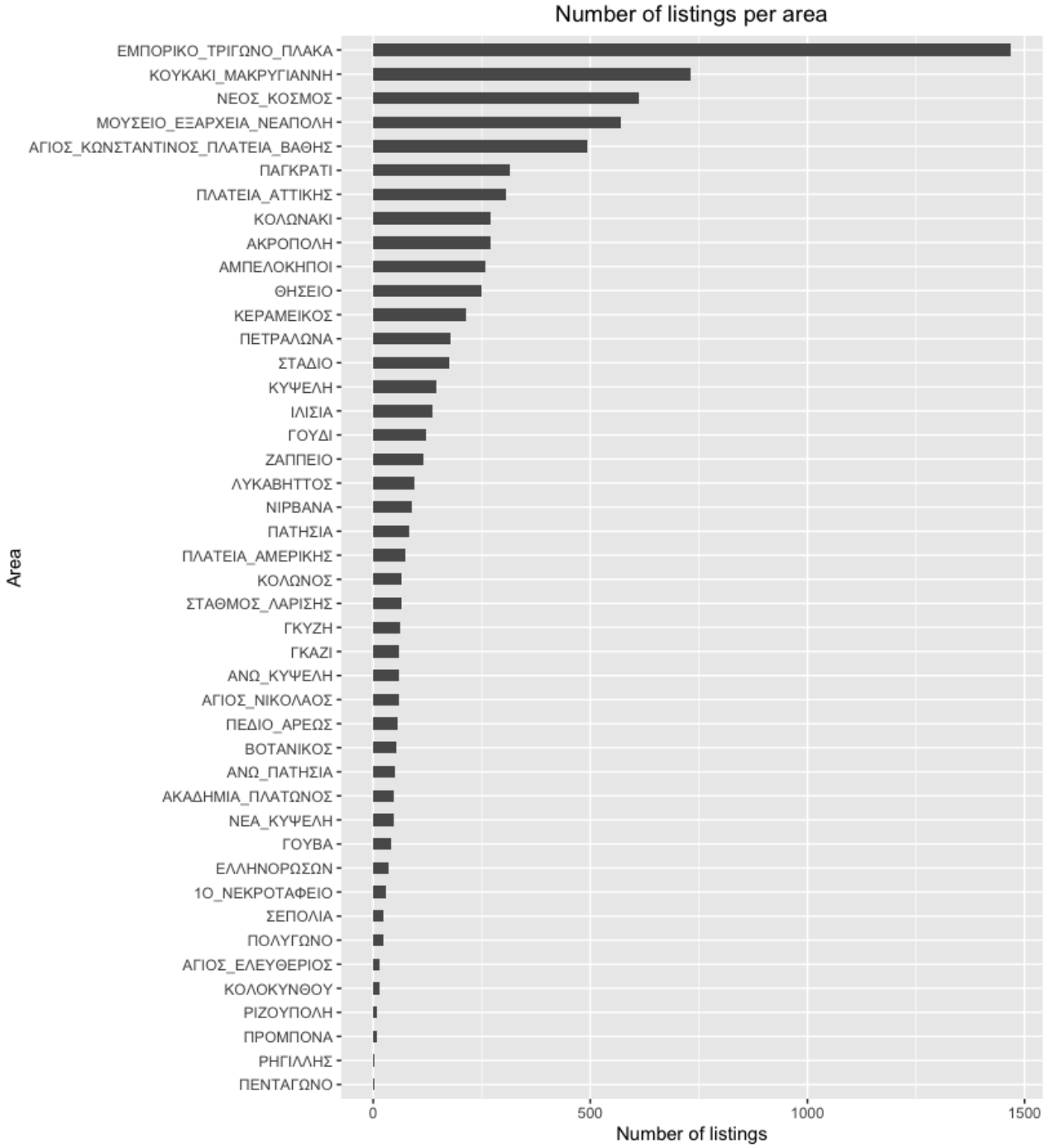
```
airbnb_2$bs.log_price <- log10(airbnb_2$bs.price)
```

- Εφαρμογή PCA στις 183 μεταβλητές κειμενικής ανάλυσης (tf). Επιλογή 33ων PCA components με αθροιστική διακύμανση - πληροφορία 80%.

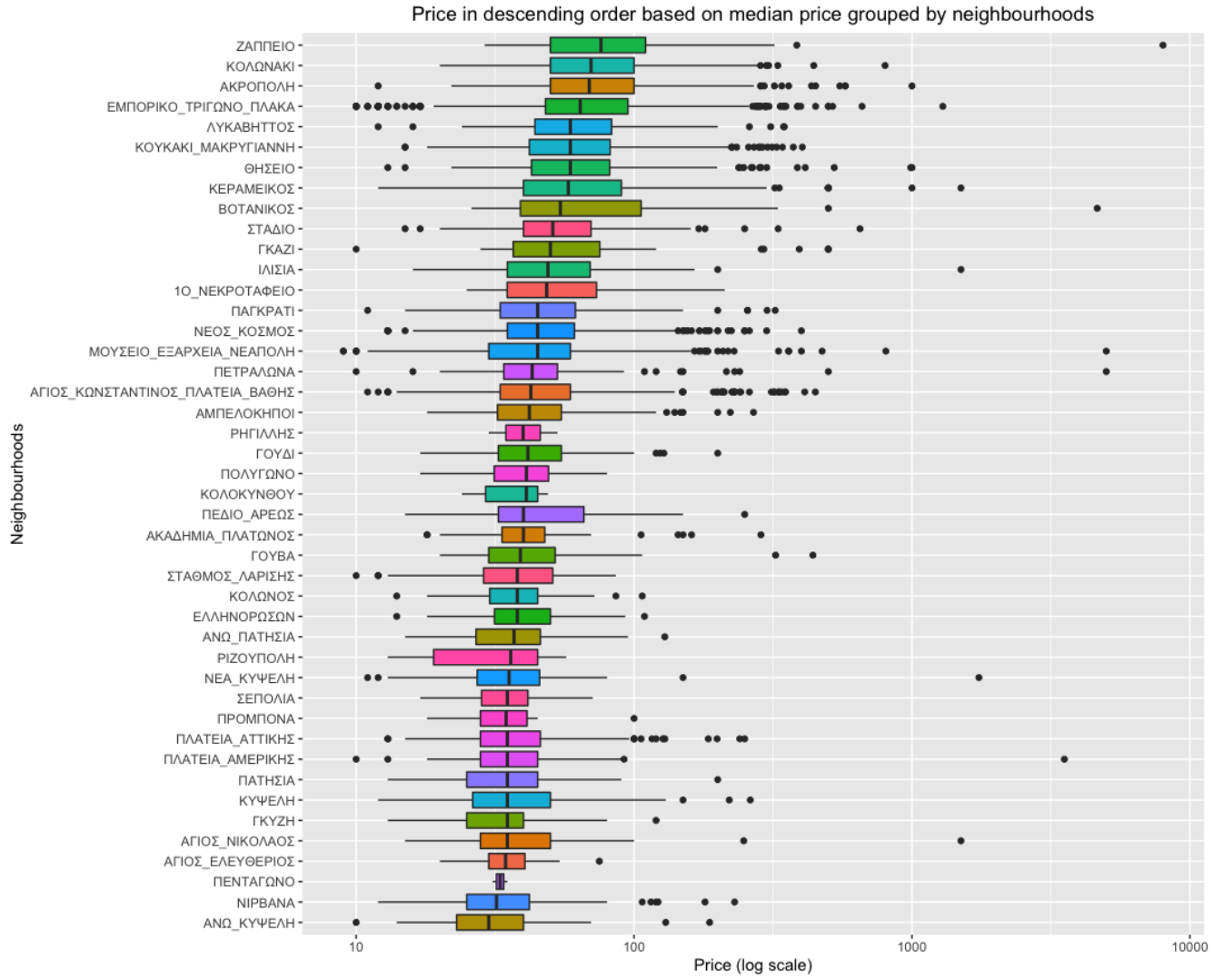
```
# Select 183 tf variables
pca_airbnb_tf <- select(airbnb_2, starts_with('tf'))
# Create components
myPr <- prcomp(pca_airbnb_tf)
# Examine components with threshold 0.8
summary(myPr)
# Select 33 components with cumulative proportion and bind to airbnb_2
airbnb_2 <- cbind(airbnb_2, myPr$x[,1:33])
```

- Δημιουργία 2 συνόλων δεδομένων με κατηγορικές μεταβλητές και dummy μεταβλητές
- Αφαίρεση της μεταβλητής τιμή από το σύνολο δεδομένων μιας και υπάρχει η λογαριθμοποιημένη μεταβλητή.

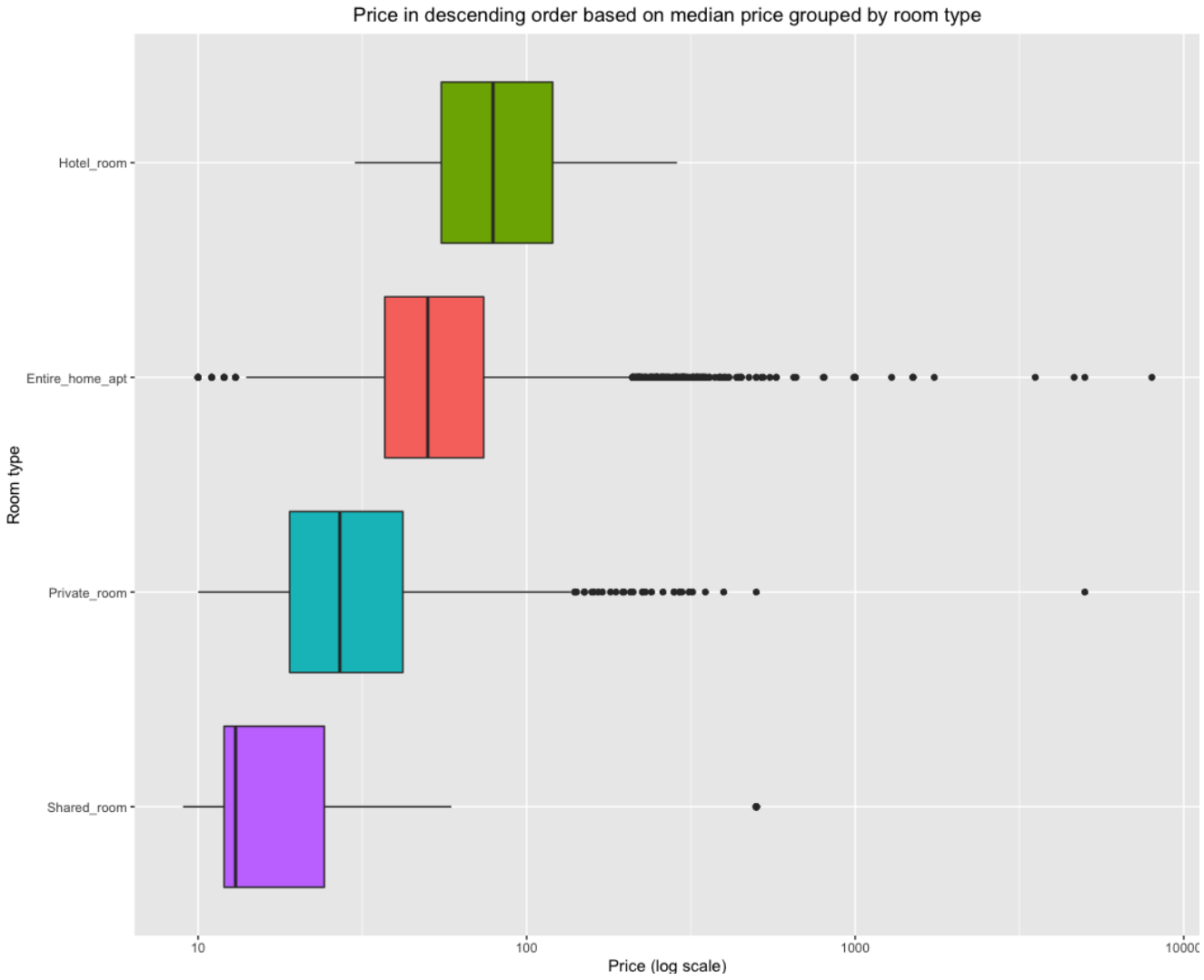
# Οπτικοποίηση δεδομένων



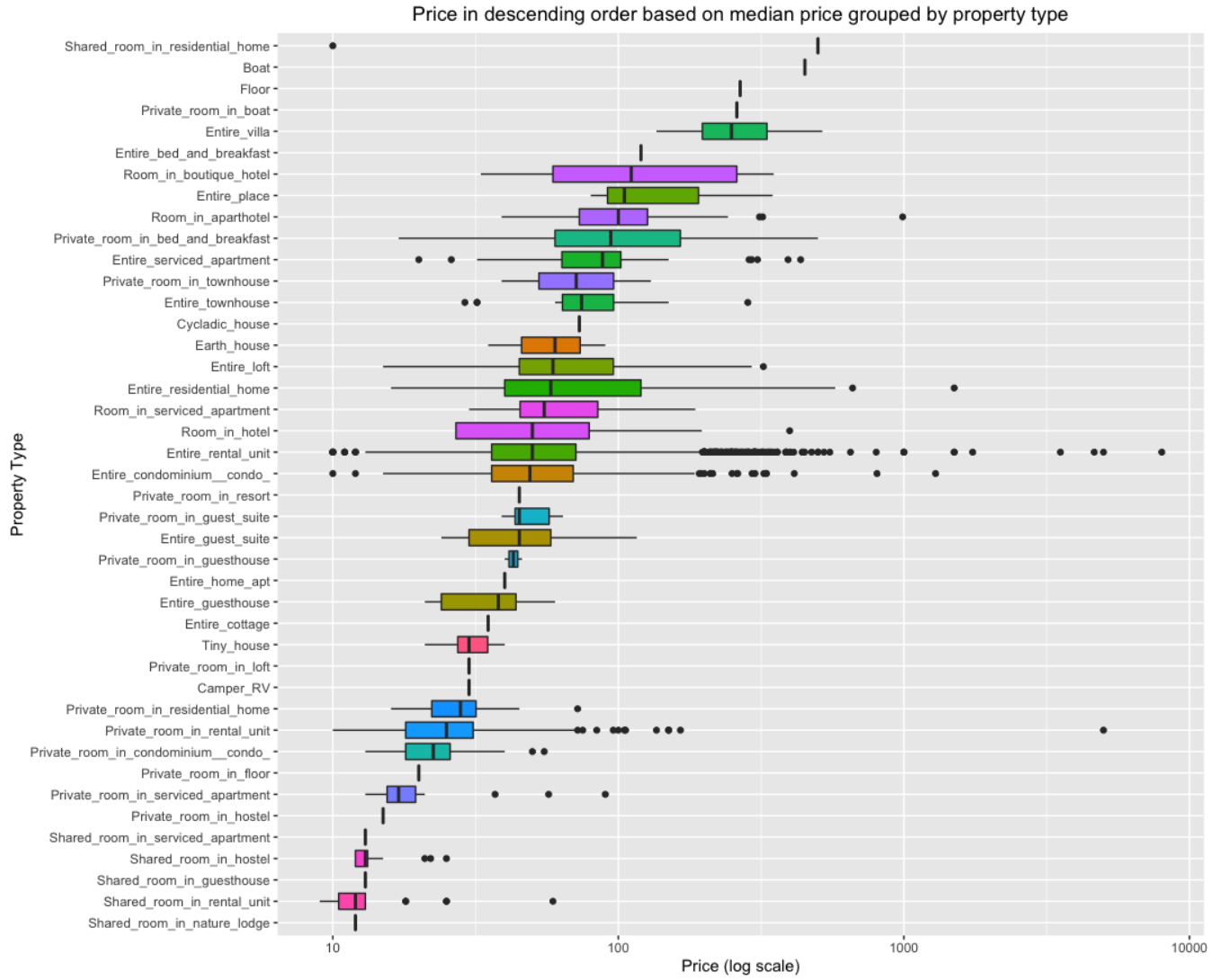

Διάγραμμα με το άθροισμα των καταλυμάτων ανά περιοχή. Διακρίνεται πως η περιοχή του κέντρου συγκεντρώνει το μεγαλύτερο αριθμό καταλυμάτων.



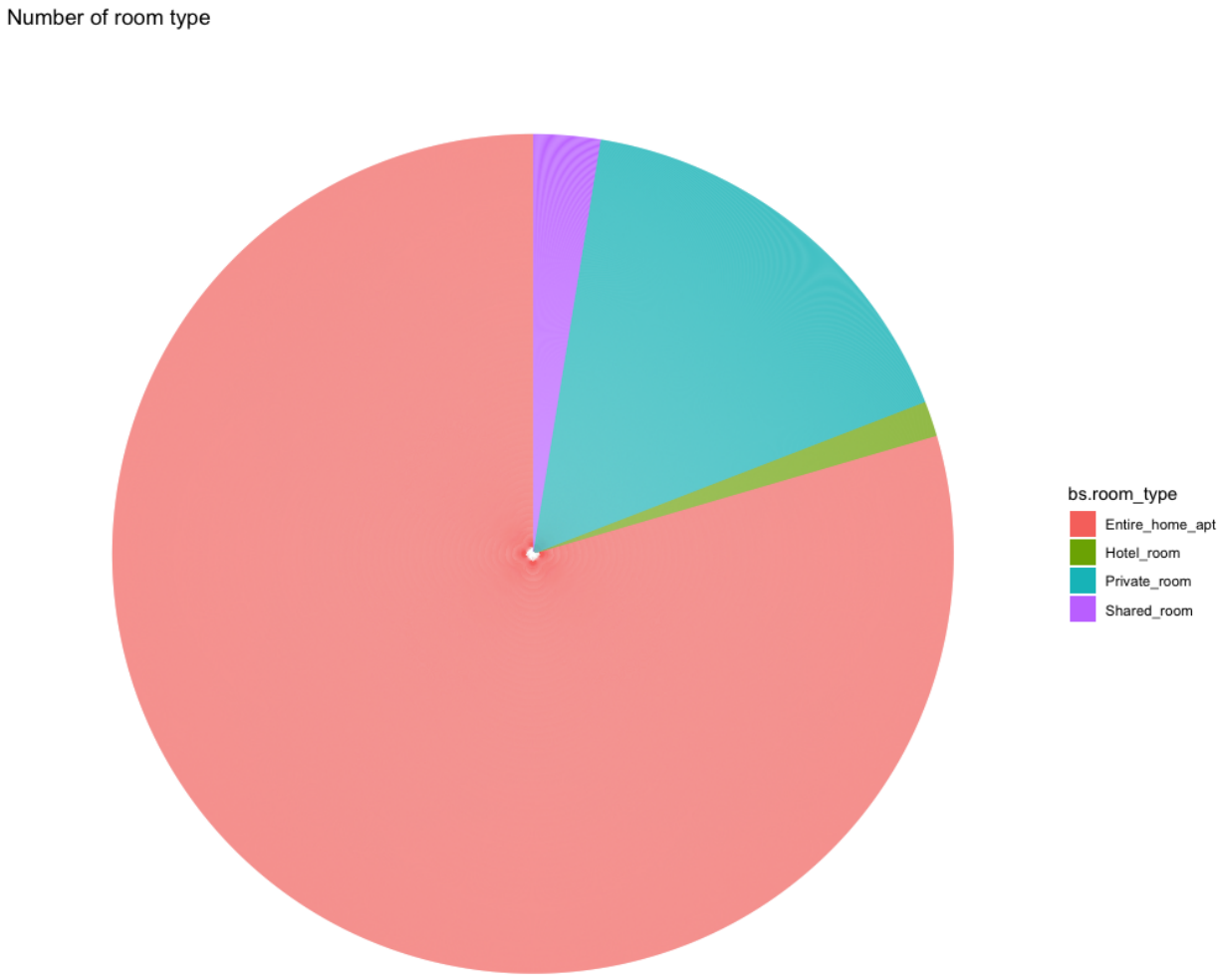

Διάγραμμα παρουσίασης των περιοχών κατά φθίνουσα σειρά με βάση την διάμεση τιμή. Διακρίνονται οι περιοχές Ζάππειο, Κολωνάκι, Ακρόπολη, Εμπορικό Τρίγωνο Πλάκα να έχουν τις μεγαλύτερες τιμές συγκριτικά με τις υπόλοιπες περιοχές. Η μεγαλύτερη τιμή του συνόλου δεδομένων παρουσιάζεται στο Ζάππειο και στην συνέχεια στον Βοτανικό, Νέο Κόσμο, Μουσείο - Εξάρχεια - Νεάπολη.



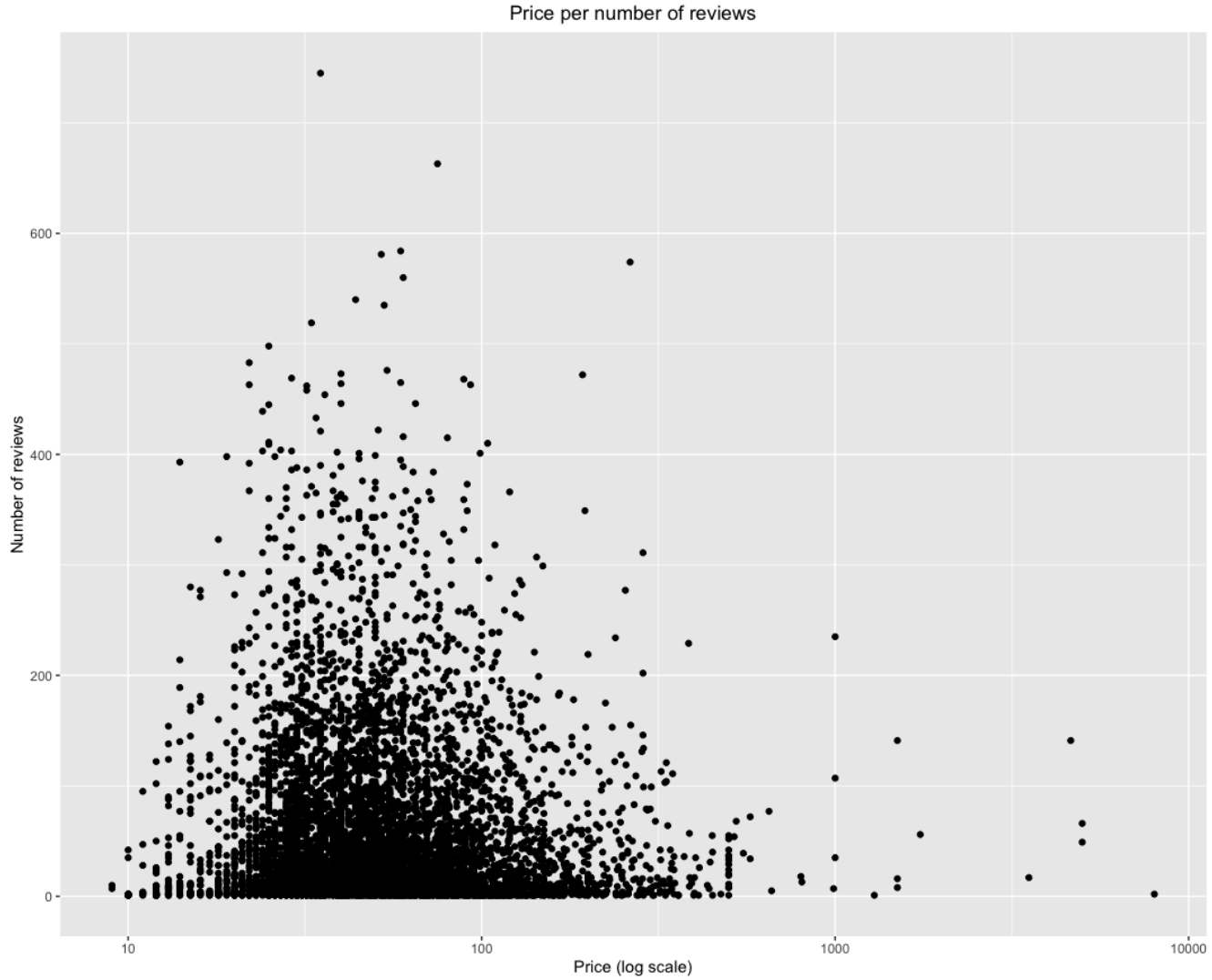

Στο δεύτερο διάγραμμα παρουσιάζεται η τιμή του κάθε καταλύματος σε φθίνουσα σειρά σύμφωνα με την διάμεση τιμή. Παρατηρούμε ότι ο πιο ακριβός τύπος δωματίου είναι τα Hotel Rooms με πολύ υψηλές τιμές που φτάνουν σχεδόν τα 500 ευρώ, χωρίς να υπάρχουν ακραία τιμές, ενώ ακολουθεί ο τύπος δωματίου Entire_home_apt και Private_room με παρόμοιες τιμές και εμφάνιση πολλών ύποπτων ή ακραίων τιμών. Στα shared rooms παρατηρούνται λογικές τιμές με μια παρατήρηση να είναι outlier.



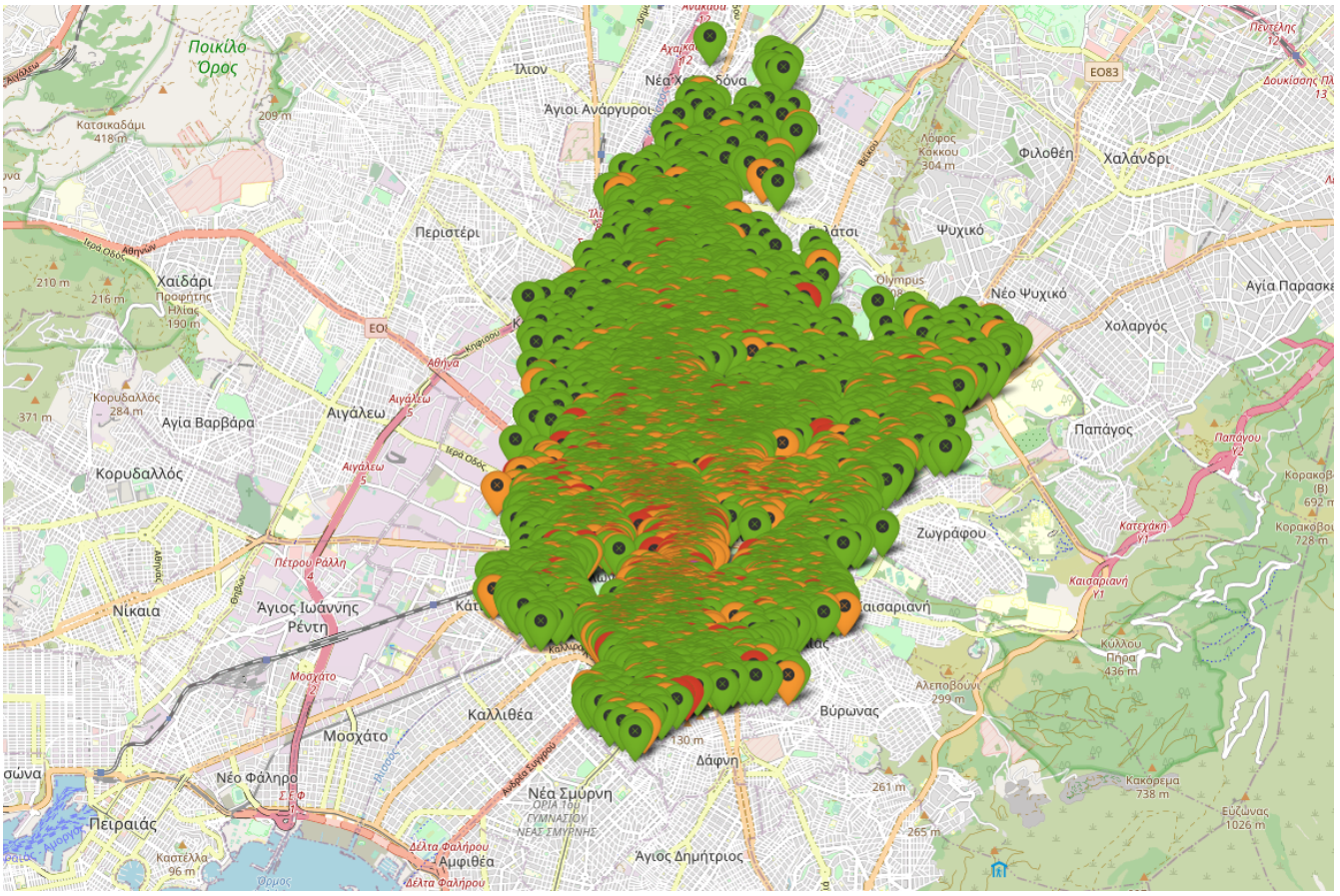

Στο συγκεκριμένο box plot παρουσιάζονται οι τιμές των καταλυμάτων σε φθίνουσα σειρά (σύμφωνα με την διάμεσο) σε σχέση με τον τύπο της ιδιοκτησίας των καταλυμάτων. Παρατηρούμε πως ακριβότερα είναι τα καταλύματα-βίλες, δωμάτια σε ξενοδοχεία boutique και ολόκληρο σπίτι. Αντίθετα χαμηλότερη τιμή παρουσιάζουν τα καταλύματα που είναι κοινόχρηστα δωμάτια , hostels και ιδιωτικά δωμάτια σε κοινόχρηστα διαμερίσματα. Έντονη ποικιλομορφία στη τιμή των καταλυμάτων εντοπίζεται στα ολόκληρα διαμερίσματα καθώς πολλές από τις τιμές βρίσκονται έξω από το ενδοτεταρτημοριακό εύρος.




Στο παρόν διάγραμμα απεικονίζονται οι τέσσερις τύποι δωματίων των καταλυμάτων. Είναι φανερό πως το μεγαλύτερο ποσοστό των καταλυμάτων είναι τύπου διαμέρισμα, ολόκληρο σπίτι. Σχεδόν ισάριθμα και σε μικρό πλήθος εμφανίζονται τα δωμάτια ξενοδοχείου και τα κοινόχρηστα δωμάτια. Τέλος περίπου το ένα πέμπτο του ποσοστού των καταλυμάτων αποτελούν τα καταλύματα με ιδιωτικό δωμάτιο.




Στο παραπάνω scatter plot παρουσιάζεται ο αριθμός των αξιολογήσεων σε σχέση με την τιμή των καταλυμάτων σε δεκαδική λογαριθμική κλίμακα. Σύμφωνα με το plot μπορούμε να συμπεράνουμε πως τα ακριβά καταλύματα δεν διαθέτουν πολλές αξιολογήσεις. Όπως επίσης ότι η πλειοψηφία των καταλυμάτων έχουν ελάχιστες έως και καθόλου αξιολογήσεις. Τέλος πολλές αξιολογήσεις, άνω των 100, εντοπίζονται σε καταλύματα με τιμή από 20-120 ευρώ.




Απαραίτητη θεωρήθηκε και η δημιουργία ενός διαδραστικού χάρτη στον οποίο εντοπίζεται η ακριβή διεύθυνση των καταλυμάτων σε σχέση με τον πραγματικό χώρο, Επίσης έγινε μια χρωματική ομαδοποίηση των καταλυμάτων σύμφωνα πάντα με τις τιμές αυτών. Με πράσινο χρώμα απεικονίζονται τα καταλύματα που έχουν τιμή από 30 μέχρι και 70, με πορτοκαλί χρώμα τα καταλύματα που έχουν τιμή μεταξύ 70-200 ευρώ, με κόκκινο χρώμα ακριβότερα καταλύματα με τιμή μεγαλύτερη των 200 και μικρότερη των 5000 ευρώ.. και με μωβ χρώμα είναι τα εξαιρετικά καταλύματα άνω των 5000 ευρώ των τα οποία είναι και τα λιγότερα. Το πρώτο συμπέρασμα που δημιουργείται είναι πως τα περισσότερα καταλύματα στην περιοχή της Αθήνας ανήκουν στη πρώτη κατηγορία και θεωρούνται οικονομικά. Από την άλλη πλευρά τα ακριβά καταλύματα εντοπίζονται στο κέντρο της Αθήνας και ο αριθμός τους μειώνεται προς τις απομακρυσμένες από το κέντρο περιοχές. Τέλος πάνω σε κάθε πινέζα (κατάλυμα) αναφέρεται και η τιμή του καταλύματος σε ευρώ.

# Εφαρμογή μοντέλων μηχανικής μάθησης παλινδρόμησης

Στο σημείο αυτό εφαρμόστηκαν διάφορα μοντέλα μηχανικής μάθησης για την πρόβλεψη της μεταβλητής στόχου bs.log_price η οποία είναι ο δεκαδικός λογάριθμος της μεταβλητής bs.price. Τα μοντέλα που χρησιμοποιήθηκαν είναι η **Support Vector machine** η **Multiple Linear Regression** και η **Random Forest**. Το πρώτο μοντέλο SVM επεξεργάστηκε το dataset με όνομα airbnb_4 ενώ τα υπόλοιπα μοντέλα επεξεργάστηκαν το dataset με όνομα *airbnb_3.* Για την εκτέλεση των παραπάνω μοντέλων χωρίστηκε το dataset σε Train set και Test set με ποσοστό **80%** και **20%** αντίστοιχα.

## Support Vector Machine

Τα Support Vector Machine (SVM) είναι ένα εργαλείο πρόβλεψης, ταξινόμησης και παλινδρόμησης που χρησιμοποιεί τη θεωρία της μηχανικής μάθησης για τη μεγιστοποίηση της προγνωστικής ακρίβειας ενώ αυτόματα αποφεύγονται τα υπερβολικά ταιριαστά δεδομένα. Η κατηγοριοποίηση των δεδομένων στηρίζεται στην εύρεση ενός βέλτιστου υπερεπιπέδου που διαχωρίζει τα δεδομένα δημιουργώντας το μέγιστο περιθώριο. Οι SVM αποτελούν μία σύγχρονη αποτελεσματική προσέγγιση της επίλυσης ζητημάτων κατηγοριοποίησης αλλά με κατάλληλες διαφοροποιήσεις της βασικής μεθοδολογίας κατηγοριοποίησης σε δύο κλάσεις μπορούν να επιλυθούν προβλήματα περισσότερων κλάσεων, παλινδρόμησης (regression) και αναγνώρισης προτύπων.

**Εφαρμογή**

Για την εφαρμογή της μεθόδου Support Vector Machine εγκαταστάθηκε η βιβλιοθηκη *e1071.* Το dataset που χρησιμοποιήθηκε είναι το *airbnb_4* το οποίο περιέχει numeric και dummies variables και χωρίστηκε σε 80% train set και 20% test set. Για την αξιοπιστία του μοντέλου μας ενδιαφέρει η τιμή του RMSE (Root mean square error) η οποία μας δείχνει την απόκλιση που έχει κατα μέσο όρο η προβλεπόμενη τιμή του μοντέλου από τη πραγματική τιμή, όσο μικρότερη και κοντά στο μηδέν είναι η τιμή αυτή τόσο καλύτερα πρόβλεψε το μοντέλο. Το RMSE του μοντέλου Svm είναι **RMSE:0.6631** τιμή μεγάλη συγκριτικά με τα υπόλοιπα μοντέλα που εφαρμόστηκαν**.** Αναλυτικότερα υπάρχει μεγάλη απόκλιση μεταξύ των πραγματικών τιμών των καταλυμάτων και των τιμών που προέβλεψε το μοντέλο.

## Multiple Linear Regression

Το μοντέλο της πολλαπλής γραμμικής παλινδρόμησης μοιάζει πάρα πολύ με αυτό της απλής γραμμικής παλινδρόμησης μόνο που αντί για μία ανεξάρτητη μεταβλητή έχει περισσότερες. Κάθε μια από τις ανεξάρτητες μεταβλητές μπορούν να συσχετίζεται γραμμικά με την εξαρτημένη μεταβλητή, οι μεταβλητές μπορούν να είναι ποσοτικές ή κατηγορικές. Η γραμμική σχέση της πολλαπλής γραμμικής παλινδρόμησης για k ανεξάρτητες μεταβλητές, έχει τη μορφή:

$$\mathbf{Y = b_0 + b1X_1 + b2X_2 + \ldots + b_kX_k + e}$$

Όπου $b_0$ ο σταθερός όρος, $b_1, b_2, \ldots, b_k$ οι συντελεστές κλίσης που αντιστοιχούν σε κάθε ανεξάρτητη μεταβλητή και e τα κατάλοιπα.

**Εφαρμογή**

Για τη δημιουργία του μοντέλου μηχανικής μάθησης παλινδρόμησης χρησιμοποιήθηκε η βιβλιοθήκη *caret* και η βιβλιοθήκη *lattice*. Επιπλέον έγινε χρήση της επαναληπτικής διαδικασίας cross

validation 5 folds η οποία παίρνει κάθε φορά ένα διαφορετικό 20% Test set από το συνολικό dataset για να είναι πιο αντιπροσωπευτικό το δείγμα και αυτό γίνεται 5 φορές. Αρχικά τα δεδομένα που χρησιμοποιήθηκαν για τη δημιουργία του μοντέλου ήταν μόνο numeric και η τιμή του RMSE (Root mean square error) ήταν **RMSE(num) = 0.1970.** Στη συνέχεια χρησιμοποιήθηκαν όλα τα δεδομένα από το dataset (airbnb_3) και categorical και numeric με αποτέλεσμα το **RMSE(cat-num) =0.1817** να βελτιωθεί και να συνεχίσουμε την ανάλυση **χωρίς την αφαίρεση των κατηγορικών μεταβλητών**. Η σημαντικότητα των μεταβλητών απεικονίζονται παρακάτω :

| | Overall |
|---|---|
| bs.bath_sum | 100.00 |
| bs.accommodates | 81.57 |
| bs.property_typeShared_room_in_residential_home | 54.56 |
| PC3 | 50.92 |
| bs.availability_30 | 46.37 |
| PC15 | 43.02 |
| bs.number_of_reviews | 32.76 |
| bs.review_scores_cleanliness | 31.27 |
| PC11 | 30.95 |
| bs.total_amenities | 26.77 |
| bs.number_of_reviews_ltm | 25.45 |
| bs.host_is_superhost | 24.86 |
| bs.property_typeEntire_villa | 24.55 |
| bs.number_of_reviews_l30d | 24.28 |

Η μεταβλητή που σχετίζεται με τη μεταβλητή στόχο είναι το άθροισμα των μπάνιων που διαθέτει ένα κατάλυμα. Επίσης η τιμή του καταλύματος επηρεάζεται αρκετά από τους διαμένοντες και τρίτη στη σειρά μεταβλητή που επηρεάζει το bs.log_price είναι το εάν το δωμάτιο είναι κοινόχρηστο σε κοινόχρηστο κατάλυμα (π.χ. hostel). Τέλος το Principal Component 3 και 11 φαίνονται να έχουν επίδραση στη τιμή του καταλύματος.

## Random Forest

Ο random forest, είναι ένας δημοφιλής αλγόριθμος ταξινόμησης που χρησιμοποιείται για πρόβλεψη και λειτουργεί τόσο με κατηγορικές μεταβλητές (προβλήματα ταξινόμησης), όσο και με συνεχείς μεταβλητές (προβλήματα παλινδρόμησης). Είναι ένας εποπτευόμενος αλγόριθμος μηχανικής μάθησης, που κατασκευάζει ένα μεγάλο σύνολο από μεμονωμένα δέντρα απόφασης, μη συσχετισμένα μεταξύ τους, προκειμένου να πραγματοποιήσει μία πιο ακριβής και πιο σταθερή πρόβλεψη. Η μείωση της συσχέτισης μεταξύ των δέντρων, επιτυγχάνεται μέσω της επιλογής τυχαίου αριθμού μεταβλητών σε κάθε εσωτερικό κόμβο διαχωρισμού. Όσο μεγαλύτερος είναι ο αριθμός των δέντρων (το δάσος), τόσο πιο ακριβές θα είναι το αποτέλεσμα.

**Εφαρμογή**

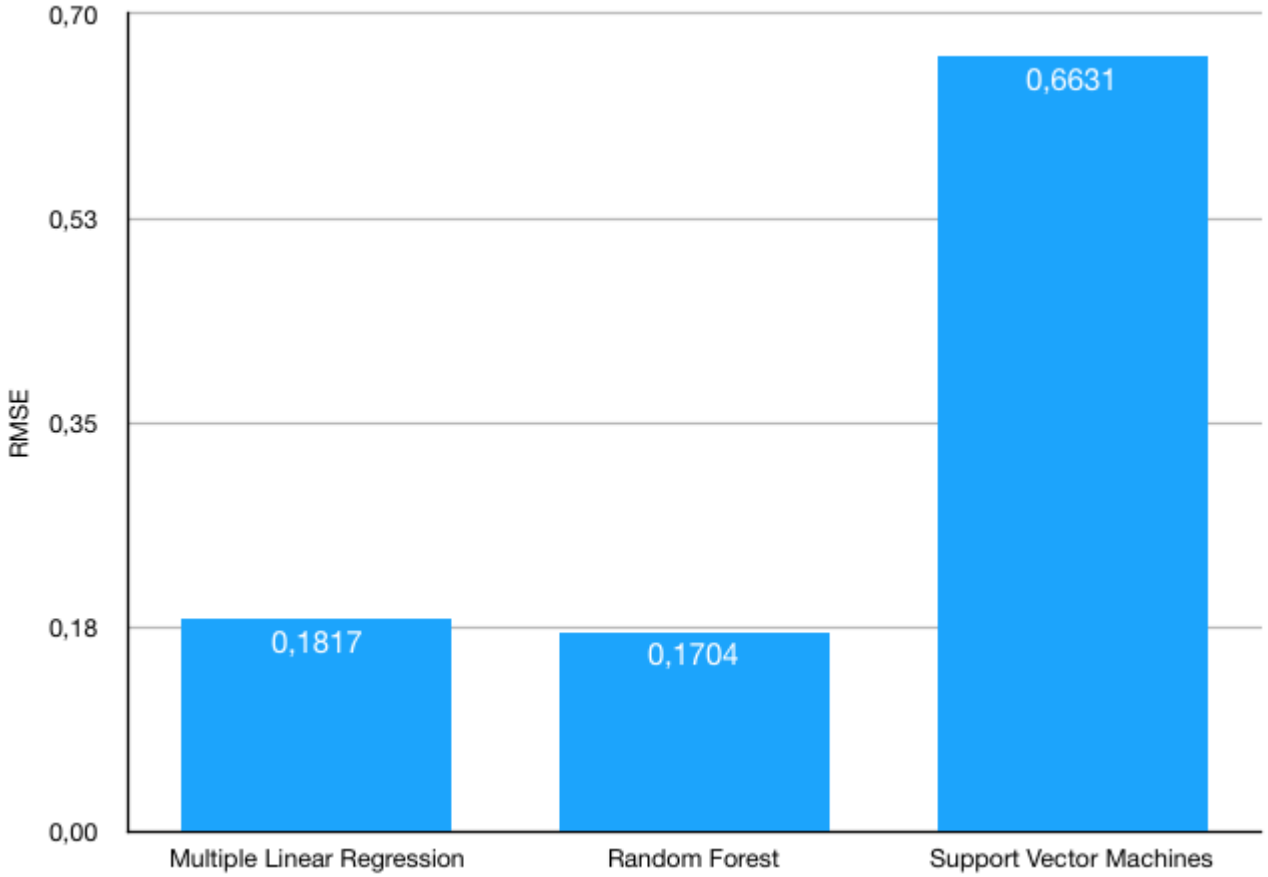
Για την εφαρμογή του μοντέλου Random Forest αναγκαία ήταν η εγκατάσταση της βιβλιοθήκης *randomForest*. Το dataset που χρησιμοποιήθηκε είναι το *airbnb_3.* Όπως και στη Multiple Regression έτσι και στη Random Forest πραγματοποιήθηκε η επαναληπτική διαδικασία cross validation 5 folds. Για τη δημιουργία του μοντέλου κατασκευάστηκαν 400 trees και η εκτέλεση του μοντέλου, όπως ήταν

λογικό, διήρκησε περισσότερο από τις υπόλοιπες. Το RMSE του μοντέλου είναι **0.1704** τιμή μικρή που μας δίνει τη δυνατότητα να καταλάβουμε πως τα αποτελέσματα της μεθόδου Random Forest είναι αρκετά κοντά στις πραγματικές τιμές. Οι σημαντικότερες μεταβλητές ακολουθούν παρακάτω :

| | Overall |
|---|---|
| bs.latitude | 100.00 |
| bs.bath_sum | 87.67 |
| bs.accommodates | 62.80 |
| bs.bath_shared | 61.96 |
| bs.host_listings_count | 48.45 |
| bs.availability_30 | 46.62 |
| PC3 | 43.03 |
| bs.review_scores_location | 39.40 |
| bs.total_amenities | 36.72 |
| bs.availability_60 | 35.42 |
| bs.longitude | 32.58 |
| bs.availability_365 | 31.94 |
| bs.review_scores_cleanliness | 31.55 |

Η μεταβλητή που επηρεάζει άμεσα την μεταβλητή στόχο bs.log_price είναι το bs.latitude ή αλλιώς η τετμημένη του καταλύματος, η τοποθεσία. Έπειτα ακολουθεί και παλι ο συνολικός αριθμός των μπάνιων του καταλύματος, στοιχείο που μας δείχνει τη σημαντικότητα της μεταβλητής bs.bath_sum και στα δυο μέχρι τώρα μοντέλα. Εξίσου σημαντικός στο να επηρεάσει τη τιμή ενός καταλύματος είναι και ο αριθμός των διαμενόντων σε κάθε κατάλυμα όπως και αν το μπάνιο είναι κοινόχρηστο. Οι υπόλοιπες μεταβλητές παρατηρούμε πως συνεισφέρουν το ίδιο στο μοντέλο καθώς οι διαφορές μεταξύ τους είναι μικρες.

# Αξιολόγηση αποτελεσμάτων - Συμπεράσματα




Διάγραμμα παρουσίασης αποτελεσμάτων μετρικής RMSE για τους 3 αλγόριθμους που δοκιμάστηκαν.

Όπως διακρίνεται από το παραπάνω διάγραμμα το μικρότερο RMSE δίνει ο αλγόριθμος Random Forest. Συγκεκριμένα RMSE (Random Forest) = 0,1704. Το RMSE ταυτίζεται με το τυπικό σφάλμα επομένως η μέτρησή του ανταποκρίνεται στην κλίμακα μέτρησης της εξαρτημένης μεταβλητής στην συγκεκριμένη περίπτωση log(price). Για να ανταποκρίνεται σε τιμές σε ευρώ πραγματοποιήθηκε η εξής εξίσωση.

Καθαρό τυπικό σφάλμα = $10^{RMSE} = 10^{0,1704} = 1,48$ ευρώ.

Αυτό δείχνει μια καλή προσαρμογή στα δεδομένα με το καθαρό τυπικό σφάλμα να κυμαίνεται στα 1,48 ευρώ.