To what extent does proximity to the underground dictate housing prices in London

I acknowledge the use of (CHATGPT(https://chat.openai.com/)) to generate materials that I have adapted to include within my final assessment. I confirm that no content generated by AI has been presented as my own work.

Introduction

London's iconic underground tube network is integral to getting around the city with a daily ridership of 3.15 million across 272 stations extending 25 miles out of London. The convenience of living near a tube station is undeniable and that's not mention any links to other rail networks such as cross country or great Anglia and the in development HS2. The convenient access to the city and the entire country, thereby potentially influencing property prices. However not all tube stations have the same links or proximity to the city centre making some stations more valuable than others which is why transport for London (TFL) has split London into 9 zones but in this report only zones one to six will be looked at as zones 7 to 9 don't encompass the entire city but only sit on the northern outskirts of London. Consequently, the first hypothesis is the closer a house is to a tube station the higher the price of the house will become, and the first null hypothesis is distance to a tube station will have zero influence on housing prices. The second hypothesis is that the effect of proximity has on prices will diminish as the station gets further away from the centre of London, and the second null hypothesis is the effect of proximity to a station will stay constant in regard to price.

Objective

The aim of this report is to provide prospective property buyers in London with insightful analysis to better understand what influences property prices. In this report the focus will be on the impact of proximity to a tube station on property prices. To start the data will have to be tidied up to ensure better accuracy and repeatability. After the data has been tidied exploratory analysis can be done using graphical and tabular representations which will guide subsequent analysis. In the analysis a linear model will firstly be fitted to give a broad overview on housing prices after which a hierarchical model can be applied to gain greater accuracy from the model and uncover any potential patterns. Understanding that most prospective property buyers aren't versed in data analytics a simple geographical map will be included to give an intuitive way to convey the information.

Data

The datasets for this analysis were curated from multiple sources. Property data, including information such as price, address and number of rooms were taken from Kaggle. Information on the tube station was also taken from Kaggle. The polygon data for the river Thames and London zones were gathered from Geofabrik and London data store, respectively. In the initial phase of data wrangling, the focus was on distilling the extensive property datasets to retain only pertinent information such as the price, address, and number of bedrooms. To obtain the geographical coordinates corresponding with the addresses, the 'tidygeocoder' library was used. This library converts address into precise coordinates through various geocoding APIs. After obtaining the coordinates of all the properties and stations, computations were done to find the nearest station for each property and the distance between them. This data was pivotal as it gave both what zone the house was in and how far away the property was to a station. The data sets were then amalgamated to contain price, address, bedrooms, distance to the station and zone. To create a geographical map, averages were taken of house prices for each zone which was then combined with the polygon data. To give context to the geographical map the river Thames was placed on top and the location of each tube station.

Analysis

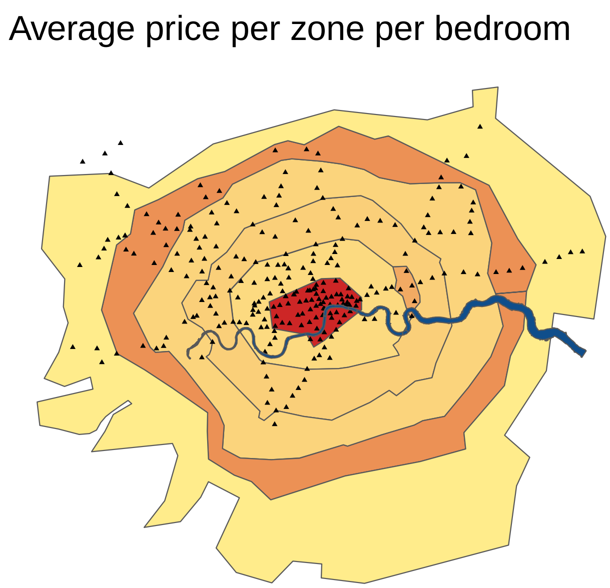


Figure 1. Average house price per bedroom subset into zones

Table 1: Averages Subset by Zone.

Zones	Price Per Bedroom(£)	Price(£)	Bedrooms	Proximity to Station	Change in Price per Bedroom(%)	Change in Price (%)
1	1079682.2	3267154	2.9	0.3	0.0	0.0
2	624128.4	2091151	3.1	0.5	-42.2	-36.0
2/3	428926.3	1353013	3.1	0.7	-60.3	-58.6
3	473781.0	2131972	4.1	0.8	-56.1	-34.7
4	453085.0	2012002	4.2	3.3	-58.0	-38.4
5	722392.1	3422629	3.8	1.6	-33.1	4.8
6	354174.8	1653823	4.4	3.2	-67.2	-49.4

It's clear that Zone 1 has an extremely high house price compared to the other zones, which is no surprise being in the middle of a capital city. It's also noticeable that the highest density of train stations is also in Zone 1, making it easier for properties to have a closer proximity to a train station. When looking at Table 1 the percentage differences show zones 2 and 5 being the closet to zone 1 prices but the other zones don't have a massive difference between them.

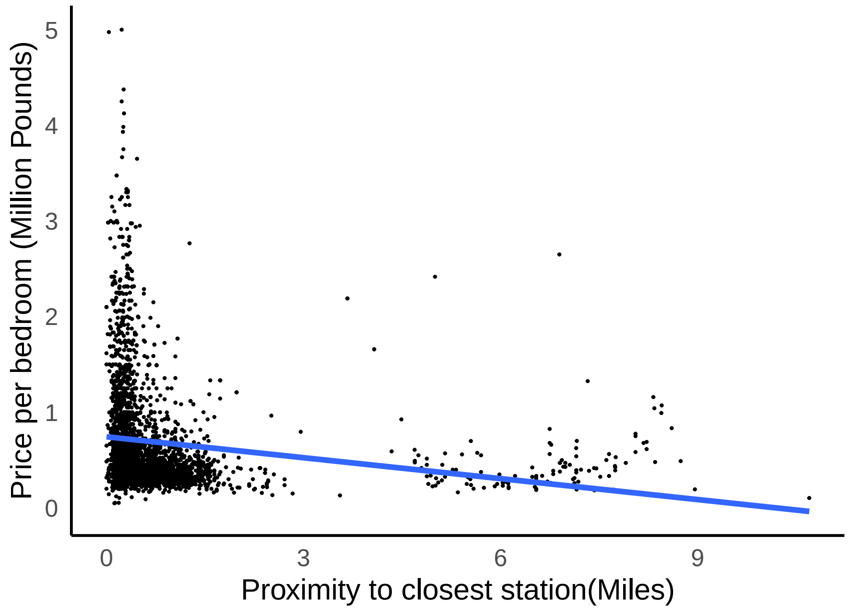


Figure 2. A linear regression plotted on the data

The linear regression indicates a pronounced downward trend as the distance increases. Although the p-values are very low(<0.05), showing that the relationship isn't due to random chance, the model fit is poor with high residuals and a very low Multiple R-squared value(0.02), meaning that the variability in price isn't adequately explained by proximity.

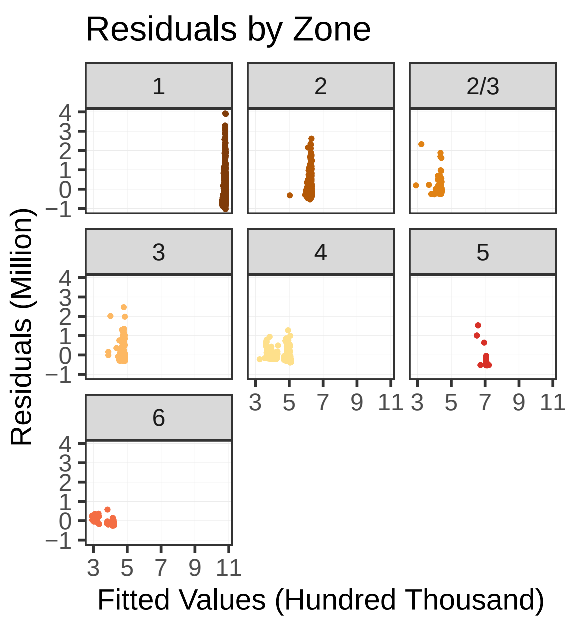


Figure 3.

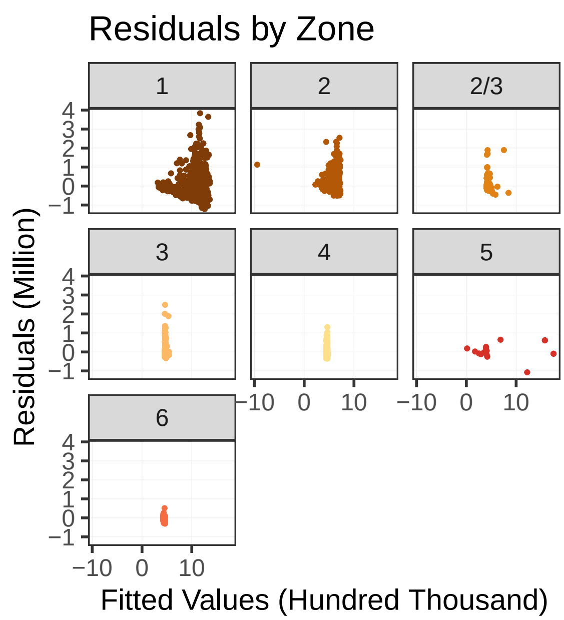
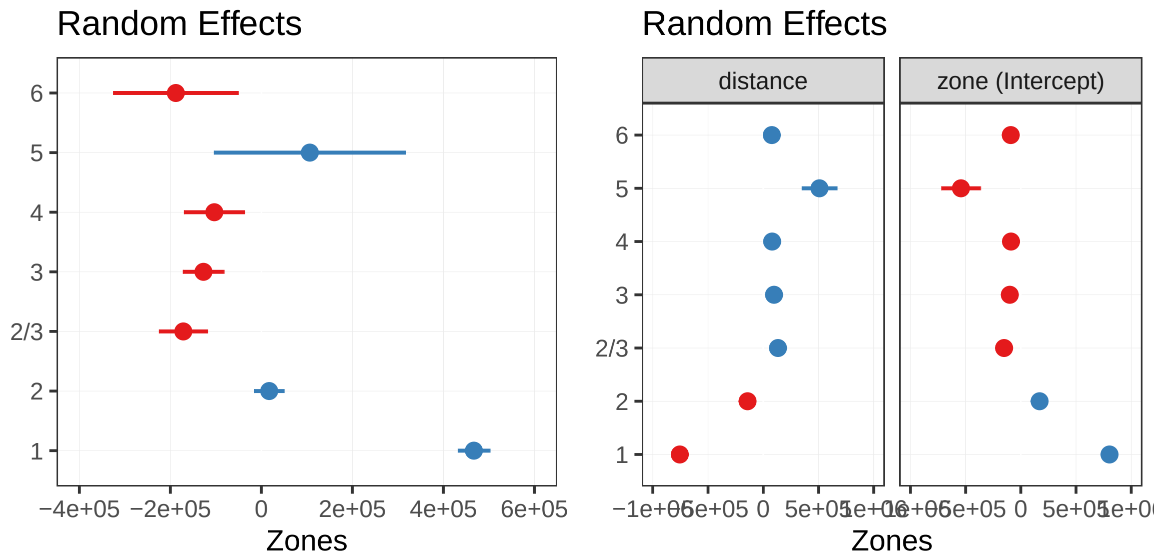


Figure 4.

The ideal scenario is for residuals to be distributed around zero; however, as you can see, neither model achieves this. Zones one, two, and four appearing to fan out as the price increases, which indicates heteroscedasticity in figure 4 which used a model with random intercepts and slopes compared to figure 3 which only had random intercepts. If heteroscedasticity is present in the data, any linear regression model will likely be a poor fit.



In both models, zones 1, 2, and 5 are outliers, with zone 1 exhibiting the greatest difference in intercepts. It appears the standard error in the second model are lower, indicating greater confidence. The standard error for zone 5 was the largest in both models, suggesting again that it could be an outlier. As zones 2/3, 2, 3, 4, and 6 are so similar it indicates these zones don't need different intercepts.

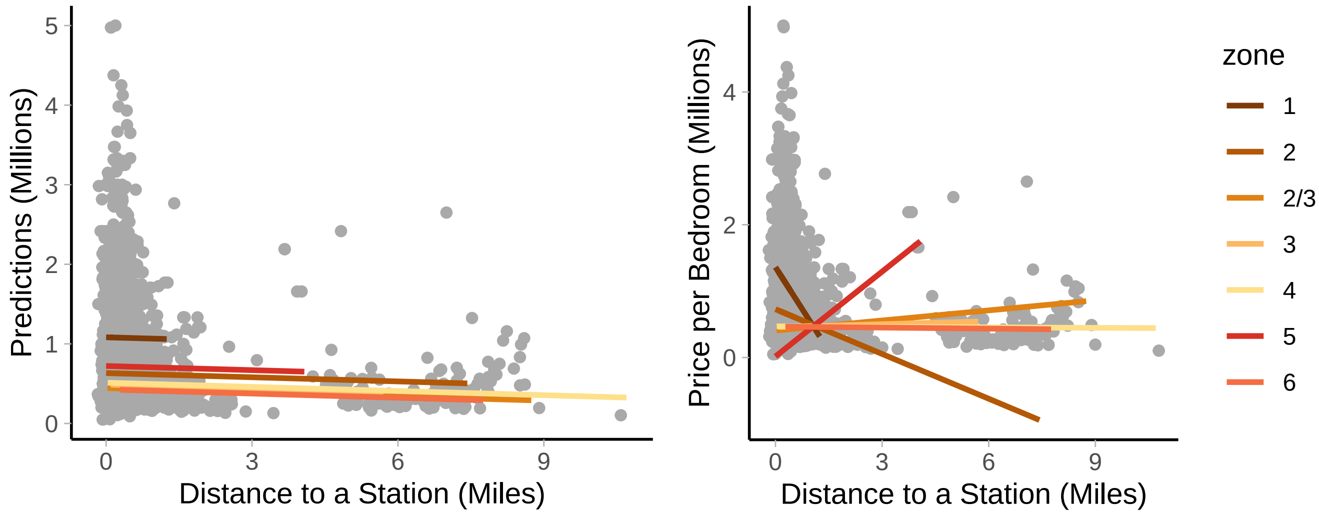


Figure 7. Model 1. Radom Slopes. *igure 8. Model 2. Random intercepts and slopes*

Comparing the two models, you can see how allowing for different slopes has significantly affected the outcomes, especially in zones 1, 2, and 5, which have repeatedly been identified as outliers. The other zones seem to maintain similar intercepts but exhibit varying slopes. The second model suggests some inaccuracies, as indicated by the prediction of negative house prices in zone 2, which may be a sign of heteroscedasticity.

Table 2: Summary statistics for the two models.

Variable	Fixed Slopes	Random Slopes
REML	86997.893	86821.669
ICC	0.170	0.492
corrolation	-0.155	-0.994

Variable	Fixed Slopes	Random Slopes
Intercept_Estimate	615791.835	557387.083
Intercept_Std.Error	93302.479	157661.183
Intercept_df	6.485	5.919
Intercept_t_value	6.600	3.535
Intercept_Pr	0.000	0.013
Distance_Estimate	-17442.252	-82803.040
Distance_Std.Error	9908.018	147301.096
Distance_df	2726.063	5.706
Distance_t_value	-1.760	-0.562
Distance_Pr	0.078	0.595
npar	4.000	6.000
AIC	87050.755	86880.364
BIC	87074.756	86916.366
logLik	-43521.377	-43434.182
deviance	87042.755	86868.364
Chisq	NA	174.391

The random slopes model is more complex as it incorporates both random intercepts and slopes, compared to the first model which includes only random intercepts. This added complexity is reflected in the larger number of parameters (npar) in the second model. Despite this added complexity, the second model has a lower Akaike Information Criterion (AIC), which helps justify the increased sophistication. Other indicators pointing to a better fit in the second model include a lower Bayesian Information Criterion (BIC), higher log-likelihood, and significant results from the ANOVA comparison of the two models. The cumulative effect of these metrics suggests that the second model may lead to more accurate and reliable predictions.

Examining the fixed effects, the intercept in the first model has a lower p-value compared to the first model, indicating increased reliability in estimating the average baseline value of price per bedroom when distance is zero. However, the random slopes model exhibits a higher standard error for the intercept, signifying increased uncertainty about the exact location of the intercept.

The fixed effect of distance shows a negative correlation with price per bedroom in the first model and a negative correlation in the second model, with neither being statistically significant. The substantially higher correlation observed in the random slopes model suggests evidence supporting the idea that zones with higher baseline ppr values tend to experience steeper declines as distance increases.

In summary, while the random slopes model is more complex, its better fit and potentially increased predictive accuracy justify the added complexity. The fixed effects suggest nuances in the relationships that are captured more effectively in the second mode

Results

Its already been seen that the linear regression had a p-value well below 0.05 but did have an ajusted r-squared value of 0.024 showing alough the p-value is low there are other varible contributing to the variation in propery price. The two hierhcial models used showed p-values above 0.05 for distance being a predictor of price however there was significant random effect amoung zones.

Table 3: Summary p-value mulltiple comparision.(Linear model had values smaller then the 18th decimal place)

Model	Original_P_Value	Bonferroni_Adjusted_P	Holm_Adjusted_P	FDR_Adjusted_P
Linear model	0.00	0.00	0.00	0.00
Fixed Slopes	0.08	0.24	0.16	0.12
Random Slopes	0.60	1.00	0.60	0.60

The initial linear model suggested that distance to the nearest tube station was a strong predictor of property prices, indicating a clear relationship between proximity and value. However, this perspective changed upon incorporating hierarchical models, which took the 'zone' factor into account.

When the hierarchical models, wherein the intercept and slope are grouped by 'zone' were applied to the data, the impact of distance seems to diminish. The lack of statistically significant evidence in these models suggests that distance alone may not be a substantial influencing factor on property prices. This implies that while being close to a tube station might generally be seen as favorable, other factors such as the specific zone in which the property is located play a more crucial role in determining its price.

Furthermore, the Intra-Class Correlation (ICC) values support this observation. The ICC essentially quantifies how strongly units in the same group resemble each other. In the context of this analysis, a low ICC would suggest that the variability within the zones is high compared to the variability between the zones, further emphasizing that the zone, rather than distance to the station, may be a more significant determinant of price.

Limitations

The data for this analysis were sourced from only three platforms, which could introduce bias. For instance, if the platforms predominantly feature either premium or lower-priced properties, the sample might not be representative of the broader market. While efforts were made to eliminate outliers, there were still noticeable extreme outliers in Zone 5.

All the models employed in this study assumed a linear relationship, which presupposes homoscedasticity. This assumption may not always hold, potentially limiting the accuracy of the predictions. Additionally, the models assumed that observations within the groups were correlated and that the residuals would follow a normal distribution.

The limited selection of variables in the analysis could introduce bias, as there may be other unaccounted factors influencing property prices. Evidence of such omitted variables was observed in the data. Although it can be challenging to include every variable, efforts should be made to mitigate this limitation. Furthermore, the possibility of multicollinearity between distance and zones raises concerns about accurately isolating their individual effects on price.

While the dataset includes over 3,000 properties, this represents only a fraction of the total properties in London. Thus, although the report is focused on London, it provides a generalized overview and may not accurately predict house prices in different locations.

To address all of these issues more data would have to be collected on multiple cities and locations and more variables should be included in the data set. For the analysis a a non linear model could be used such as logarithmic model. having the extra variables will help to prevent multicollinearity and the non linear model will prevent the heteroscedasticity.

Conclusion

The initial linear regression model suggested that proximity to a tube station is a significant factor in determining house prices. However, once hierarchical models were applied, this significance diminished.

The hierarchical models showed considerable variation in price within specific zones, particularly zones 1, 2, and 5, which remained outliers. This suggests that other variables influenced prices in these zones. For example, Zone 1, being the heart of London, may see increased prices due to more amenities, and Zone 5 may see increased prices due to specific neighborhood characteristics.

The Intraclass Correlation Coefficient (ICC) values in the hierarchical models were indicative of properties in the same zone not being as similar as the model may have assumed, suggesting that proximity isn't contributing significantly to price. The p-values also hinted towards this, implying that proximity may play a role, but isn't the sole or most influential predictive factor.

Based on the findings from the hierarchical models, it can be concluded that the evidence to support the initial hypothesis (H1) is inconclusive. While proximity to a tube station may play a role in determining property prices, it is not the sole or most influential factor. Thus, we cannot fully reject the null hypothesis (H0) but must acknowledge that the relationship between property price and proximity to a tube station is more nuanced and can be influenced by other factors, such as the specific zone and perhaps other unobserved characteristics. As the evidence for the first hypothesis was inconclusive, the second hypothesis is also inconclusive and would require more detailed variables to answer.

For prospective buyers, this analysis shows the importance of considering a multifaceted approach when evaluating house prices. While proximity to the tube may contribute to house price in some regard, it doesn't appear to be a linear relationship, and other factors seem to be significantly more influential.

Appendix

```
#load in libs
library(tidyverse)
library(dplyr)
library(ggplot2)
library(tidygeocoder)
library(lme4)
library(leaflet)
library(leaflet.extras)
library(sf)
library(magrittr)
library(sjPlot)
library(sjmisc)
library(sjstats)
library(arm)
library(lmerTest)
library(scales)
library(knitr)
#data for housing prices
df <- read_csv(file=("C:/Users/archi/Documents/LHP.CSV"))
df1 <- read_csv(file=("C:/Users/archi/Documents/LHP1.CSV"))
df2 <- read_csv(file=("C:/Users/archi/Documents/LHP2.CSV"))
#poloygon data for the zones of london and the thames
water <- st_read(dsn = "C:/Users/archi/Documents/TFL_ZONE/water.kml")
kml_data <- st_read(dsn = "C:/Users/archi/Documents/TFL_ZONE/TFL.kml")
```

```r
#data on the train stations
lus <- read_csv(file=("C:/Users/archi/Documents/LUS.csv"))
#gets rid of zones 789
kml_data <- kml_data[!(kml_data$Name %in% c("Zone 7","Zone 8", "Zone 9")), ]
#gets rid of all the other lakes and rives in london
water <- water[-c(1:431, 433:1042, 1044:2087, 2089:2290, 2292:2938, 2940:4000), ]
#checks its the same coord system
water <- st_transform(water, st_crs(kml_data))
# Create a new 'Address' column by combining the address components
df2$Address <- paste(df2$`Property Name`, df2$Location, df2$`City/County`, df2$`Postal Code`, sep = ",
#change col to right names
df <- df[, c("street", "price_pounds", "bedrooms", "bathrooms")]
df1 <- df1[, c("Address", "Price", "Bedrooms", "Bathrooms")]
df2 <- df2[, c("Address","Price","No. of Bedrooms", "No. of Bathrooms")]
ncn <- c("address", "price", "bedrooms", "bathrooms")
colnames(df) <- ncn
colnames(df1) <- ncn
colnames(df2) <- ncn
#combine
mydata <- rbind(df, df1, df2)
#just double check for any na
mydata <- mydata[!is.na(mydata$address), ]
# Geocode the addresses from the "address" column
geocoded <- mydata %>%
  geocode(address = address, method = 'osm')
#remove anything outside the m25
geocoded$lat <- ifelse(geocoded$lat > 51.25 & geocoded$lat < 51.689, geocoded$lat, NA)
geocoded$long <- ifelse(geocoded$long > -0.51 & geocoded$long < 0.28, geocoded$long, NA )
#basic map to check anything outside of london
map <- leaflet() %>%
  addTiles()
map <- map %>%
  addMarkers(data = geocoded, lng = ~long, lat = ~lat)
#add station to the name so i can get the coords
lus$Station <- paste(lus$Station, "Station")
# gets coords for stations and make sure its all within the m25
geocoded_lus <- lus %>%
  geocode(address = Station, method = 'osm')
geocoded_lus$lat <- ifelse(geocoded_lus$lat > 51.25 & geocoded_lus$lat < 51.689, geocoded_lus$lat, NA)
geocoded_lus$long <- ifelse(geocoded_lus$long > -0.51 & geocoded_lus$long < 0.28, geocoded_lus$long, NA
# Check for missing values and remove rows with missing coordinates
geocoded <- geocoded[complete.cases(geocoded[c("long", "lat")]), ]
geocoded_lus <- geocoded_lus[complete.cases(geocoded_lus[c("long", "lat")]), ]
#Changes the coords into distance by mile
property_sf <- st_as_sf(geocoded, coords = c("long", "lat"), crs = 4326)
station_sf <- st_as_sf(geocoded_lus, coords = c("long", "lat"), crs = 4326)
property_sf_utm <- st_transform(property_sf, crs = 32633)
station_sf_utm <- st_transform(station_sf, crs = 32633)
distances_m <- st_distance(property_sf_utm, station_sf_utm)
distances_mi <- distances_m / 1609.34
# Find the index of the closest station for each property
closest_station_index <- apply(distances_mi, 1, which.min)
# Create a data frame with the results
```

```r
closest_df <- data.frame(PropertyID = geocoded$address,
                         Closest_StationID = geocoded_lus$Station[closest_station_index],
                         distance = apply(distances_mi, 1, min))
#bind them so i can then search for station info
bind <- cbind(geocoded[, c("address", "price", "bedrooms", "bathrooms", "lat", "long")],
              closest_df[, c("Closest_StationID", "distance")])
# Find matching values and inserts to bind data set
matches <- lus$Station[lus$Station %in% bind$Closest_StationID]
bind$station_info <- ifelse(bind$Closest_StationID %in% matches, lus$Station[match(bind$Closest_Station
bind$line <- ifelse(bind$Closest_StationID %in% matches, lus$`Line(s)`[match(bind$Closest_StationID, lus
bind$usage <- ifelse(bind$Closest_StationID %in% matches, lus$`Usage (millions per year)`[match(bind$Cl
bind$zone <- ifelse(bind$Closest_StationID %in% matches, lus$`Zone(s)`[match(bind$Closest_StationID, lus
#takes out anomalies
threshold <- 2
mean_value <- mean(bind$price)
std_value <- sd(bind$price)
anomalies <- bind$price > (mean_value + threshold * std_value) | bind$price < (mean_value - threshold *
bind <- bind[!anomalies, ]
#seperating character data.
bind$zone <- gsub("2 & 3", "2/3", bind$zone)
bind <- bind %>%
  separate_rows(zone, sep = ' & ')
#making price per bedroom col
bind$ppr <- bind$price / bind$bedrooms
#not looking at zone 7 8 or 9 and zero bedrooms isnt possible
bind <- bind %>%
  filter(bedrooms > 0)%>%
  filter(zone < 7)
#creates averages table
averages <- bind %>%
  group_by(zone)%>%
  summarise(across(where(is.numeric), mean))
averages <- averages[, c("zone", "ppr", "price", "bedrooms", "distance")]
#giving the polygon data averges of the zones
result <- cbind(kml_data,averages)
# Create a customized color scale for the heat map
color_scale <- scale_fill_gradient(
  low = "lightgoldenrod1",
  high = "firebrick3",
  limits = c(min(result$ppr), max(result$ppr)),
  breaks = seq(min(result$ppr), max(result$ppr), length.out = 5)
)
heat_map <- ggplot(data = result) +
  geom_sf(aes(fill = ppr)) +
  geom_point(data = geocoded_lus, aes(x=long, y=lat), color = "black", size = 0.4, shape = 17) +
  geom_sf(data = water, fill = "dodgerblue4", alpha = 1) +
  color_scale +
  labs(title = "Average price per zone per bedroom", fill = "Price in Pounds",
       caption = expression(italic("Figure 1. Average house price per bedroom subset into zones"))) +
  theme_minimal() +
  theme(axis.title = element_blank(),
        axis.text = element_blank(),
        panel.grid = element_blank(),
```

```
        axis.ticks = element_blank(),
        legend.position = "none")
#simple linear regression
dflm <- lm(ppr~distance, data = bind)
dflm_plot <- ggplot(bind, aes(x=distance, y=ppr)) +
  geom_point(size = 0.05) +
  geom_smooth(method = "lm", se = FALSE) +
  labs(x="Proximity to closest station(Miles)", y = "Price per bedroom (Million Pounds)", caption = exp
  scale_y_continuous(labels = function(x) x / 1e6) +
  theme_minimal() +
  theme(
    panel.grid.major = element_blank(),
    panel.grid.minor = element_blank(),
    axis.line = element_line(colour = "black")
  )
#random intercepts
dflmer1 <- lmer(ppr~distance + (1|zone), data = bind, control = lmerControl(optimizer = "Nelder_Mead"))
#random slopes
dflmer2 <- lmer(ppr~distance + (1+distance|zone), data = bind, control = lmerControl(optimizer = "Nelde
performance::icc(dflmer1)
dflmer1 %>% fixef() %>% round(5)
dflmer1 %>% se.coef() %>% extract2(1) %>% round(5)
dflmer1diag <- data.frame(residuals=resid(dflmer1),
                         zone=bind$zone,
                         Fitted=fitted(dflmer1))
custom_colors <- c("#7f3b08", "#b35806", "#e08214", "#fdb863", "#fee08b", "#d73027", "#f46d43")
llr <- ggplot(data=dflmer1diag, aes(x=Fitted, y=residuals, col=zone)) +
  geom_point(size = 0.4) +
  facet_wrap(~zone) +
  scale_y_continuous(labels = function(x) x / 1e6) +
  scale_x_continuous(labels = function(x) x / 1e5) +
  scale_color_manual(values = custom_colors) +
  labs(x = "Fitted Values (Hundred Thousand)", y = "Residuals (Million)", caption = expression(italic("
  ggtitle("Residuals by Zone") +
  theme_bw() +
  theme(
    panel.grid.major = element_line(size = 0.1),
    panel.grid.minor = element_blank(),
    legend.position = "none"  #
  )
re  <- plot_model(dflmer1, type = "re")
re <- re +
  theme_bw() +
  theme(
    panel.grid.major = element_line(size = 0.1),
    panel.grid.minor = element_blank()
  ) +
  labs(
    title = "Random Effects",
    y = "Zones",
  )
dflmer2diag <- data.frame(residuals=resid(dflmer2),
                          zone=bind$zone,
```

```r
                          Fitted=fitted(dflmer2))
llr2 <- ggplot(data=dflmer2diag, aes(x=Fitted, y=residuals, col=zone)) +
  geom_point(size = 0.4) +
  facet_wrap(~zone) +
  scale_y_continuous(labels = function(x) x / 1e6) +
  scale_x_continuous(labels = function(x) x / 1e5) +
  scale_color_manual(values = custom_colors) +
  labs(x = "Fitted Values (Hundred Thousand)", y = "Residuals (Million)", caption = expression(italic("F
  ggtitle("Residuals by Zone") +
  theme_bw() +
  theme(
    panel.grid.major = element_line(size = 0.1),
    panel.grid.minor = element_blank(),
    legend.position = "none"
  )
re2 <- plot_model(dflmer2, type = "re")
re2 <- re2 +
  theme_bw() +
  theme(
    panel.grid.major = element_line(size = 0.1),
    panel.grid.minor = element_blank()
  ) +
  labs(
    title = "Random Effects",
    y = "Zones")
bind$dflmer1.predictions <- predict(dflmer1)
bind$dflmer2.predictions <- predict(dflmer2)
#plots model
pr <- ggplot(bind, aes(x = distance, y = dflmer1.predictions)) +
  geom_jitter(aes(y = ppr), color = "darkgray", width = 0.2, height = 0.2, size = 1.5) +
  geom_line(aes(color = zone), size = 1) +
  scale_color_manual(values = c("#7f3b08", "#b35806", "#e08214", "#fdb863", "#fee08b", "#d73027", "#f46
  scale_y_continuous(labels = function(x) x / 1e6) +
  labs(x = "Distance to a Station (Miles)",y = "Predictions (Millions)", caption = expression(italic("Fi
  theme_light() +
  theme(
    panel.grid.major = element_blank(),
    panel.grid.minor = element_blank(),
    panel.border = element_blank(),
    axis.line.x = element_line(color = "black"),
    axis.line.y = element_line(color = "black"),
    legend.position = "none"  # Remove legend
  )
pr2 <- ggplot(bind, aes(x = distance, y = dflmer2.predictions)) +
  geom_jitter(aes(y = ppr), color = "darkgray", width = 0.2, height = 0.2, size = 1.5) +
  geom_line(aes(color = zone), size = 1) +
  scale_color_manual(values = c("#7f3b08", "#b35806", "#e08214", "#fdb863", "#fee08b", "#d73027", "#f46
  scale_y_continuous(labels = function(x) x / 1e6) +
  labs(x="Distance to a Station (Miles)", y = "Price per Bedroom (Millions)", caption = expression(itali
  theme_light() +
  theme(
    panel.grid.major = element_blank(),
    panel.grid.minor = element_blank(),
```

```r
    panel.border = element_blank(),
    axis.line.x = element_line(color = "black"),
    axis.line.y = element_line(color = "black")
  )
#creating tables
averages$PDP <- (averages$ppr - averages$ppr[1]) / averages$ppr[1] * 100
averages$PDPPR <- (averages$price - averages$price[1]) / averages$price[1] * 100
tab <- kable(averages,
             digits = 1,
             col.names = c("Zones", "Price Per Bedroom(£)", "Price(£)", "Bedrooms", "Proximity to Stati
              caption = "Averages Subset by Zone."
            )
anova_table <- anova(dflmer1, dflmer2)
model1 <- summary(dflmer1)
model2 <- summary(dflmer2)
reml1 <- model1$AICtab["REML"]
reml2 <- model2$AICtab["REML"]
fe1 <- model1$coefficients
fe2 <- model2$coefficients
model1_intercept <- c(Estimate = fe1["(Intercept)", "Estimate"],
                      Std.Error = fe1["(Intercept)", "Std. Error"],
                      df = fe1["(Intercept)", "df"],
                      t_value = fe1["(Intercept)", "t value"],
                      Pr = fe1["(Intercept)", "Pr(>|t|)"])
model1_distance <- c(Estimate = fe1["distance", "Estimate"],
                     Std.Error = fe1["distance", "Std. Error"],
                     df = fe1["distance", "df"],
                     t_value = fe1["distance", "t value"],
                     Pr = fe1["distance", "Pr(>|t|)"])

model2_intercept <- c(Estimate = fe2["(Intercept)", "Estimate"],
                      Std.Error = fe2["(Intercept)", "Std. Error"],
                      df = fe2["(Intercept)", "df"],
                      t_value = fe2["(Intercept)", "t value"],
                      Pr = fe2["(Intercept)", "Pr(>|t|)"])
model2_distance <- c(Estimate = fe2["distance", "Estimate"],
                     Std.Error = fe2["distance", "Std. Error"],
                     df = fe2["distance", "df"],
                     t_value = fe2["distance", "t value"],
                     Pr = fe2["distance", "Pr(>|t|)"])
combined_fe_table <- data.frame(
  Model = c("Model 1", "Model 2"),
  Intercept_Estimate = c(model1_intercept["Estimate"], model2_intercept["Estimate"]),
  Intercept_Std.Error = c(model1_intercept["Std.Error"], model2_intercept["Std.Error"]),
  Intercept_df = c(model1_intercept["df"], model2_intercept["df"]),
  Intercept_t_value = c(model1_intercept["t_value"], model2_intercept["t_value"]),
  Intercept_Pr = c(model1_intercept["Pr"], model2_intercept["Pr"]),
  Distance_Estimate = c(model1_distance["Estimate"], model2_distance["Estimate"]),
  Distance_Std.Error = c(model1_distance["Std.Error"], model2_distance["Std.Error"]),
  Distance_df = c(model1_distance["df"], model2_distance["df"]),
  Distance_t_value = c(model1_distance["t_value"], model2_distance["t_value"]),
  Distance_Pr = c(model1_distance["Pr"], model2_distance["Pr"])
)
```

```r
icc_res2 <- performance::icc(dflmer2)
icc_res1 <- performance::icc(dflmer1)
vcov_matrix <- as.matrix(model1$vcov)
cor_matrix <- cov2cor(vcov_matrix)
print(cor_matrix)
cor_df <- as.data.frame(as.table(cor_matrix))
colnames(cor_df) <- c("Variable1", "Variable2", "Correlation")
second_row <- cor_df[2, ]
vcov_matrix2 <- as.matrix(model2$vcov)
cor_matrix2 <- cov2cor(vcov_matrix2)
cor_df2 <- as.data.frame(as.table(cor_matrix2))
colnames(cor_df2) <- c("Variable1", "Variable2", "Correlation")
second_row2 <- cor_df2[2, ]
corrfe <- rbind(second_row, second_row2)
numtab <- data.frame(
  Model = c("Model 1", "Model 2"),
  REML = c(reml1, reml2),
  ICC = c(icc_res1$ICC_adjusted, icc_res2$ICC_adjusted)
)
numtab$corrolation <- corrfe$Correlation
combined_fe_table <- combined_fe_table[, !(names(combined_fe_table) %in% c("Model"))]
final_table <- cbind(numtab, combined_fe_table, anova_table)
final_table_long <- final_table %>%
  pivot_longer(cols = -Model, names_to = "Variable", values_to = "Value") %>%
  pivot_wider(names_from = Model, values_from = Value)
final_table_long <- final_table_long[1:19,]
final_table_long <- final_table_long %>%
  rename(
    Variable = Variable,
    `Fixed Slopes` = `Model 1`,
    `Random Slopes` = `Model 2`
  )
numtab <- kable(final_table_long, digits = 3, caption = "Summary statistics for the two models.")
p_value_model1 <- summary(dflm)$coefficients["distance", "Pr(>|t|)"]
p_value_model2 <- summary(dflmer1)$coefficients["distance", "Pr(>|t|)"]
p_value_model3 <- summary(dflmer2)$coefficients["distance", "Pr(>|t|)"]
p_values <- c(p_value_model1, p_value_model2, p_value_model3)
p.adjust(p_values, method = "bonferroni")
p.adjust(p_values, method = "holm")
p.adjust(p_values, method = "fdr")
adjusted_p_bonferroni <- p.adjust(p_values, method = "bonferroni")
adjusted_p_holm <- p.adjust(p_values, method = "holm")
adjusted_p_fdr <- p.adjust(p_values, method = "fdr")
adjusted_p_table <- data.frame(
  Model = c("Linear model", "Fixed Slopes", "Random Slopes"),
  Original_P_Value = c(round(p_values[1], 3), round(p_values[2:3], 2)),
  Bonferroni_Adjusted_P = c(round(adjusted_p_bonferroni[1], 3), round(adjusted_p_bonferroni[2:3], 2)),
  Holm_Adjusted_P = c(round(adjusted_p_holm[1], 3), round(adjusted_p_holm[2:3], 2)),
  FDR_Adjusted_P = c(round(adjusted_p_fdr[1], 3), round(adjusted_p_fdr[2:3], 2))
)
ptab <- kable(adjusted_p_table, caption = "Summary p-value mulltiple comparision.(Linear model had valu
```

chatgpt

dflmer3diag <- data.frame(residuals=resid(dflmer3), + zone=df$zone, +Fitted = fitted(dflmer3))Errorindata.frame(residu$
$resid(dflmer3), zone = df$zone, Fitted = fitted(dflmer3)) : arguments imply differing number of rows: 3146, 3154 ChatGPT The error message you're seeing, "arguments imply differing number of rows," indicates that the objects you are trying to combine into a data frame have a different number of rows. In your case, the resid(dflmer3) has 3146 rows, while df$zone and fitted(dflmer3) have 3154 rows each.

To fix this issue, you need to ensure that all the vectors you are combining into the data frame have the same number of rows. You can do this by checking the data and identifying the source of the discrepancy. Here are a few things you can try:

Check the structure of your data frame (df) to ensure it has the expected number of rows. You can use the nrow(df) function to determine the number of rows in your data frame.

Verify that the model object dflmer3 has the same number of rows as your data frame. You can use the nobs(dflmer3) function to check the number of observations in the model.

Examine any data cleaning or preprocessing steps you have applied to your data before fitting the model to make sure there are no discrepancies in row numbers.

Once you identify the source of the discrepancy, you can take appropriate actions to ensure that all vectors have the same number of rows before creating your data frame.

User is it justified to compare a linear model adn a hierchcial model ChatGPT Comparing a linear model and a hierarchical model is justified in specific situations, depending on your research objectives and the nature of your data. Both types of models serve different purposes and are suitable for different types of problems.

Linear Model:

Linear models are used to describe the relationship between one or more independent variables (predictors) and a dependent variable (response) assuming a linear relationship. They are appropriate when the data can be adequately described by a simple linear equation and when there is no inherent structure in the data due to hierarchy or grouping. Linear models are often used for simple regression analysis, analysis of variance (ANOVA), and multiple regression. Hierarchical Model (also known as a mixed-effects or multilevel model):

Hierarchical models are used when your data has a nested or hierarchical structure, such as repeated measurements within individuals, data clustered within groups, or random effects due to different levels of grouping. They allow for modeling the variability at multiple levels and can account for correlation and heterogeneity within and between groups. Hierarchical models are particularly useful for dealing with clustered or longitudinal data, where observations are not independent. When deciding whether to use a linear model or a hierarchical model, consider the following factors:

Data Structure: If your data exhibits a hierarchical or nested structure, where observations are not independent within groups or levels, a hierarchical model is often more appropriate.

Research Objectives: Think about the questions you want to answer. If you are interested in understanding both the overall population-level effects and the individual-level variations, a hierarchical model is more suitable.

Model Complexity: Linear models are simpler and more interpretable. If your data doesn't have a hierarchical structure and doesn't require accounting for random effects, a linear model might suffice.

Available Software and Expertise: Implementing hierarchical models may require specialized statistical software (e.g., Stan, BUGS, or R packages like lme4). You should also have the necessary expertise to set up and interpret hierarchical models.

In some cases, it might be valuable to compare both types of models to assess which one better fits the data and answers your research questions. You can use statistical criteria (e.g., AIC, BIC, or likelihood ratio tests) to compare model fit and decide which model is more appropriate for your specific analysis.

User when making a hierical model should i come it to another hierichal model or should i just compare it to a linear model ChatGPT When deciding whether to develop a hierarchical model, it's important to consider your research objectives and the structure of your data. Whether you should compare a hierarchical model to another hierarchical model or to a linear model depends on the specific characteristics of your data and the questions you want to answer. Here are some guidelines:

Comparing Hierarchical Models to Linear Models:

If your data has a clear hierarchical or nested structure, and you want to assess the impact of this structure on your outcome, then it's appropriate to compare a hierarchical model to a linear model. Linear models assume independence of observations, which may not be valid in hierarchical data. A hierarchical model allows you to account for the within-group correlation or variability in a more appropriate way. Comparing Hierarchical Models to Other Hierarchical Models:

If your data has multiple levels or sources of variability, you may consider comparing different hierarchical models to assess which one best fits your data and research objectives. You might compare models with different random effects structures, varying in terms of complexity. This can help you identify the most appropriate level of hierarchy for your data. The choice between comparing a hierarchical model to a linear model or to another hierarchical model depends on the complexity of your data and the goals of your analysis. Generally, it's good practice to start with a basic hierarchical model to capture the known hierarchical structure in your data. Once you have established this baseline model, you can explore whether adding more levels, different random effects, or interaction terms improves the model's fit and helps answer your research questions.

Additionally, you can use statistical criteria such as AIC (Akaike Information Criterion), BIC (Bayesian Information Criterion), or likelihood ratio tests to compare different models and select the one that provides the best trade-off between fit and complexity. Comparing different hierarchical models can be particularly valuable when your data involves complex nesting or varying sources of heterogeneity.

User in the data the range near 0 is large but as the data gets further from zero the range gets smaller. what model would you apply here ChatGPT When your data exhibits a range that is large near zero and gets smaller as it moves away from zero, you are dealing with heteroscedasticity. Heteroscedasticity is a violation of the assumption of constant variance in linear regression models. To address this issue, you can consider the following modeling approaches:

Transform the Data: One common way to address heteroscedasticity is to transform the dependent variable. You can try different transformations, such as taking the logarithm or square root of the response variable, which can stabilize the variance. After the transformation, you can fit a linear model to the transformed data.

Weighted Least Squares (WLS): In some cases, you can use weighted least squares regression, where you assign weights to each data point to account for the varying variances. Data points with larger variances receive smaller weights, and those with smaller variances receive larger weights. This method is useful when the heteroscedasticity structure is known.

Robust Regression: Robust regression methods, such as Huber regression or robust standard errors, can be used to account for heteroscedasticity and outliers in the data. These methods provide more robust parameter estimates in the presence of non-constant variance.

Generalized Linear Models (GLM): If the relationship between the mean and variance of the data is non-linear, you might consider using a generalized linear model with an appropriate family and link function. This allows you to model the relationship more flexibly.

Non-Linear Regression: If your data exhibits a clear pattern that suggests a non-linear relationship between the predictor and the response variable, you can fit a non-linear regression model. Non-linear models can better capture the changing variance structure.

Data Transformation + Hierarchical Model: In some cases, hierarchical models (mixed-effects models) can be useful to account for heteroscedasticity due to data collected from different sources or groups. You can combine data transformation with a hierarchical model to address both non-constant variance and group-specific effects.

The choice of which method to use depends on the specific characteristics of your data, the underlying assumptions, and the goals of your analysis. It's important to assess the effectiveness of these approaches through diagnostic checks and model comparisons to ensure that you are adequately addressing the heteroscedasticity issue in your data.

User summary(dflm)

Call: lm(formula = price ~ Distance, data = df)

Residuals: Min 1Q Median 3Q Max -2001266 -1277076 -709986 345742 7448653

Coefficients: Estimate Std. Error t value Pr(>|t|)
(Intercept) 2167246 46049 47.064 < 2e-16 ***Distance -111762 28795 -3.881 0.000106*** — Signif. codes: 0 '' ***0.001*** '' *0.01* '' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1986000 on 2859 degrees of freedom Multiple R-squared: 0.005241, Adjusted R-squared: 0.004893 F-statistic: 15.06 on 1 and 2859 DF, p-value: 0.0001063 ChatGPT The output you provided is the summary of a linear regression model (lm) in R. Here's an interpretation of the key information in this summary:

Call: This line shows the formula used to fit the linear regression model. In this case, the model was fitted with the formula "price ~ Distance" using the "df" dataset.

Residuals: This section provides statistics about the residuals (the differences between the observed values and the values predicted by the model).

Min: The minimum residual value is -2,001,266. 1Q: The first quartile (25th percentile) of residuals is -1,277,076. Median: The median (50th percentile) of residuals is -709,986. 3Q: The third quartile (75th percentile) of residuals is 345,742. Max: The maximum residual value is 7,448,653. Coefficients: This section provides information about the estimated coefficients in the model.

The intercept (Intercept): The estimated intercept is 2,167,246, with a standard error of 46,049. This means that when the distance is zero, the estimated price is 2,167,246. Distance: The estimated coefficient for the "Distance" predictor variable is -111,762, with a standard error of 28,795. This coefficient represents the change in price associated with a one-unit change in distance. The negative coefficient suggests that, on average, as the distance increases, the price tends to decrease. Significance Levels: The "Pr(>|t|)" column provides the p-values associated with each coefficient. In this case, both the intercept and the Distance coefficient have very low p-values, indicating that they are statistically significant. The "***" next to the coefficient estimates indicates that they are highly significant (p-value < 0.001).

Residual Standard Error: The residual standard error is a measure of the variability of the residuals around the regression line. In this case, it is approximately 1,986,000.

Multiple R-squared and Adjusted R-squared: These statistics measure the goodness of fit of the model.

Multiple R-squared: It is a measure of the proportion of the variance in the dependent variable (price) that is explained by the independent variable (Distance). In this case, the model explains only 0.5241% of the variance in price. Adjusted R-squared: It is a version of R-squared that adjusts for the number of predictors in the model. In this case, it is slightly lower at 0.4893%. F-statistic and p-value: The F-statistic tests the overall significance of the model. It tests whether at least one of the predictors is useful in explaining the variability in the dependent variable. The low p-value (0.0001063) suggests that the model is statistically significant as a whole.

In summary, this linear regression model attempts to predict the "price" variable based on the "Distance" predictor variable. The Distance coefficient is statistically significant and suggests that, on average, as the distance increases, the price tends to decrease. However, the model's R-squared value is quite low, indicating that the model explains only a small proportion of the variance in the price.

User tell me more about the residuals ChatGPT Residuals in a linear regression model represent the differences between the observed values (actual data points) and the values predicted by the regression model. Analyzing residuals is an essential part of assessing the model's goodness of fit and checking whether the model assumptions are met. Here's more information about the residuals from the summary output you provided:

Min, 1Q, Median, 3Q, Max: These statistics provide a summary of the distribution of residuals:

Min: The minimum residual value is -2,001,266. This is the smallest negative difference between the observed and predicted values. 1Q (First Quartile): The first quartile (25th percentile) of residuals is -1,277,076. This means that 25% of the residuals fall below this value. Median (50th Percentile): The median is -709,986. It represents the middle value of the residuals when they are sorted in ascending order. 3Q (Third Quartile): The third quartile (75th percentile) of residuals is 345,742. This means that 75% of the residuals fall below this value. Max: The maximum residual value is 7,448,653. This is the largest positive difference between the observed and predicted values. Residual Standard Error: The residual standard error (also known as the root mean square error, RMSE) is a measure of the typical magnitude of the residuals. In your model, it is approximately 1,986,000. A lower residual standard error indicates a better fit of the model to the data.

Interpretation of Residuals: The residuals should ideally be normally distributed with a mean of zero and constant variance. Deviations from these assumptions may indicate model misspecification. Here are some interpretations of your residuals:

The minimum residual of -2,001,266 is quite large, indicating a significant negative deviation from the predicted values for at least one data point. The positive maximum residual of 7,448,653 suggests a substantial positive deviation from the predicted values for another data point. The distribution of residuals may be right-skewed, with a longer tail of positive residuals. Residual Analysis: You can further analyze the residuals by creating residual plots, such as a histogram or a Q-Q plot, to check for normality and homoscedasticity. If the residuals show a clear pattern in these plots, it might indicate issues with the model's assumptions.

Outliers: Large residuals, such as the minimum and maximum values you have, could indicate potential outliers or influential data points. It's essential to investigate these points to determine if they are legitimate or if they should be addressed.

In summary, the information about the residuals in your linear regression model provides insights into the distribution of errors between the observed and predicted values. Careful analysis of residuals helps assess the model's fit and identify potential issues that may require further investigation or model improvement.

User but im looking at housing price whihc cant go to zero ChatGPT You are correct that in the context of housing prices, it's not meaningful for the price to go to zero. In the case of linear regression, the assumption of normally distributed residuals with a mean of zero and constant variance may not hold perfectly for housing price data. Housing prices typically have a lower bound (they cannot be negative), and the distribution of residuals may not be normal.

When dealing with housing price data or other similar situations, where the response variable has a natural lower bound and is typically positively skewed, you might consider alternative modeling approaches. Here are a few options:

Non-linear Models: Instead of a linear regression model, you can use non-linear regression models that are more appropriate for bounded response variables. Common choices include generalized linear models (GLMs), such as Poisson or Gamma regression, or more flexible models like generalized additive models (GAMs).

Data Transformation: You can apply data transformations to the response variable to address non-constant variance or non-normality in the residuals. For example, you can log-transform the housing prices before fitting the model.

Robust Regression: Robust regression methods can handle outliers and deviations from normality. They provide more robust parameter estimates when dealing with non-normally distributed data.

Hierarchical Models: If your data involves multiple levels or clusters (e.g., houses within neighborhoods), hierarchical models (mixed-effects models) can be beneficial. These models can account for the hierarchical structure and help handle varying variances at different levels.

Residual Analysis: Regardless of the model choice, it's crucial to carefully analyze the residuals to identify potential issues. Outliers and non-constant variance should be addressed appropriately.

Censored Data Models: In some situations, you may be dealing with censored data, where prices are only observed within a certain range (e.g., a minimum and maximum price). In such cases, Tobit regression or survival analysis models might be appropriate.

The choice of modeling approach depends on the specific characteristics of your housing price data and the research questions you want to address. It's important to consider the unique properties of your data and select a model that best accommodates these properties while providing meaningful insights.

User summary(dflm)

Call: lm(formula = price ~ Distance * bedrooms, data = df)

Residuals: Min 1Q Median 3Q Max -2984369 -955904 -424630 230499 8157738

Coefficients: Estimate Std. Error t value Pr(>|t|)
(Intercept) 213389 99555 2.143 0.0322 *
Distance -448837 78002 -5.754 9.63e-09 ***bedrooms 684047 28468 24.029 < 2e-16*** Distance:bedrooms 37633 17629 2.135 0.0329 *
— Signif. codes: 0 '***' 0.001*** '*' *0.01* '' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1721000 on 2857 degrees of freedom Multiple R-squared: 0.2537, Adjusted R-squared: 0.2529 F-statistic: 323.7 on 3 and 2857 DF, p-value: < 2.2e-16 ChatGPT The output you provided is the summary of a linear regression model in R with an interaction term. Here's an interpretation of the key information in this summary:

Call: This line shows the formula used to fit the linear regression model. In this case, the model was fitted with the formula "price ~ Distance * bedrooms" using the "df" dataset. The "*" symbol between "Distance" and "bedrooms" indicates the interaction between these two variables.

Residuals: This section provides statistics about the residuals (the differences between the observed values and the values predicted by the model).

Min: The minimum residual value is -2,984,369. This is the smallest negative difference between the observed and predicted values. 1Q (First Quartile): The first quartile (25th percentile) of residuals is -955,904. This means that 25% of the residuals fall below this value. Median (50th Percentile): The median is -424,630. It represents the middle value of the residuals when they are sorted in ascending order. 3Q (Third Quartile): The third quartile (75th percentile) of residuals is 230,499. This means that 75% of the residuals fall below this value. Max: The maximum residual value is 8,157,738. This is the largest positive difference between the observed and predicted values. Coefficients: This section provides information about the estimated coefficients in the model, including main effects and interaction effects.

Intercept (Intercept): The estimated intercept is 213,389, with a standard error of 99,555. This is the estimated price when both "Distance" and "bedrooms" are zero. Distance: The estimated coefficient for the "Distance" predictor variable is -448,837, with a standard error of 78,002. This coefficient represents the change in price associated with a one-unit change in distance. bedrooms: The estimated coefficient for the "bedrooms" predictor variable is 684,047, with a standard error of 28,468. This coefficient represents the change in price associated with a one-unit change in the number of bedrooms. Distance:bedrooms: The estimated coefficient for the interaction between "Distance" and "bedrooms" is 37,633, with a standard error of 17,629. This coefficient represents the change in the effect of "Distance" on price for each additional bedroom. Significance Levels: The "Pr(>|t|)" column provides the p-values associated with each coefficient. In this case, some coefficients have low p-values, suggesting that they are statistically significant. The number of asterisks (*) indicates the level of significance, with more asterisks indicating higher significance.

Residual Standard Error: The residual standard error is a measure of the typical magnitude of the residuals. In your model, it is approximately 1,721,000.

Multiple R-squared and Adjusted R-squared: These statistics measure the goodness of fit of the model.

Multiple R-squared: It is a measure of the proportion of the variance in the dependent variable (price) that is explained by the independent variables in the model. In this case, the model explains 25.37% of the variance in price. Adjusted R-squared: It is a version of R-squared that adjusts for the number of predictors in the model. In this case, it is slightly lower at 25.29%. F-statistic and p-value: The F-statistic tests the overall significance of the model. The low p-value (p-value: $< 2.2e-16$) suggests that the model as a whole is statistically significant.

In this model, you are exploring the relationship between "price" and "Distance" and "bedrooms," including their interaction. The coefficients and their significance levels indicate the effect of each variable and the interaction on the price of the housing units. The R-squared values suggest that this model explains a moderate portion of the variance in housing prices.

User

2 / 2

what does the 1721000 mean for residual standard error ChatGPT The residual standard error (RSE) of 1,721,000 is a measure of the typical magnitude of the residuals (the differences between the observed values and the predicted values) in the linear regression model. It quantifies the amount of unexplained variation in the response variable (in this case, "price") that remains after fitting the regression model.

In your context, a residual standard error of 1,721,000 means that, on average, the actual housing prices deviate from the predicted values by approximately 1,721,000 currency units (e.g., dollars or other relevant currency). This value is a measure of the spread or dispersion of the residuals around the regression line.

A smaller residual standard error indicates that the model's predictions are, on average, closer to the actual values, which suggests a better fit of the model to the data. Conversely, a larger residual standard error indicates that the model's predictions have more variability, which implies a less precise fit to the data.

It's important to interpret the residual standard error in the context of your specific data and research question. Lowering the residual standard error by improving the model's fit can lead to better predictions and a more accurate understanding of the relationship between the predictor variables and the response variable.

User so if i have a RES of 1.7 million its not a great fit then ChatGPT A residual standard error (RSE) of 1.7 million, by itself, doesn't necessarily indicate whether the fit of your linear regression model is good or not. Whether an RSE of 1.7 million is considered a good fit depends on the context of your data, the scale of the response variable ("price" in your case), and the specific goals of your analysis.

Here are some considerations when interpreting the RSE:

Relative Scale: The RSE should be interpreted in the context of the scale of the response variable. If your housing prices are on the order of millions or more, an RSE of 1.7 million may represent a relatively small percentage of the price, indicating a reasonably good fit. However, if your housing prices are in a different range, 1.7 million may be a more substantial proportion of the price.

Domain Knowledge: Your knowledge of the domain and the specific problem you are trying to solve is crucial. If you know that the housing market typically exhibits a high degree of price variability, then an RSE of 1.7 million may be acceptable.

Comparison: It can be helpful to compare the RSE to the range of the response variable. If the RSE is much smaller than the range of the housing prices, it suggests that the model is explaining a significant portion of the variation. If the RSE is similar in magnitude to the range, it suggests that there is a substantial amount of unexplained variation.

Model Objectives: Consider the goals of your analysis. If the primary purpose of your model is prediction, a smaller RSE is desirable. If the model is used for hypothesis testing or understanding the relationships between variables, the focus may be on statistical significance and effect sizes rather than the RSE.

Residual Analysis: It's important to examine residual plots, such as a histogram of residuals and a Q-Q plot, to assess the normality and homoscedasticity of the residuals. Deviations from these assumptions can affect the interpretation of the RSE.

Ultimately, the suitability of a model depends on the research context and the specific problem you are trying to address. A lower RSE indicates a better fit and more precise predictions, but what constitutes an acceptable RSE varies from one application to another. It's often a trade-off between the complexity of the model and the amount of variability explained by the model.

User

2 / 2

summary(dflmer1) Linear mixed model fit by REML. t-tests use Satterthwaite's method ['lmerModLmerTest'] Formula: price ~ Distance + (1 | zone) Data: df

REML criterion at convergence: 90715.2

Scaled residuals: Min 1Q Median 3Q Max -1.4090 -0.5909 -0.3008 0.2167 4.0113

Random effects: Groups Name Variance Std.Dev. zone (Intercept) 2.857e+11 534477 Residual 3.687e+12 1920277 Number of obs: 2856, groups: zone, 7

Fixed effects: Estimate Std. Error df t value Pr(>|t|)
(Intercept) 1.878e+06 2.223e+05 6.521e+00 8.450 9.55e-05 *** Distance -5.748e+04 3.599e+04 5.038e+03 -1.597 0.11
— Signif. codes: 0 '' ***0.001*** '' *0.01* '' 0.05 '.' 0.1 ' ' 1

Correlation of Fixed Effects: (Intr) Distance -0.229 ChatGPT The output you provided is a summary of a linear mixed-effects model (LMM) fitted to your data. Linear mixed-effects models are used when your data has a hierarchical or clustered structure, where observations are grouped into different levels. In your model, you are modeling "price" as a function of "Distance" with random intercepts for the grouping variable "zone." Here's an interpretation of the key information in this summary:

Formula: The formula specifies the model you fitted. In this case, you have a fixed effect for "Distance" and a random intercept for "zone," which is indicated by (1 | zone).

REML Criterion at Convergence: This is the value of the residual sum of squares based on the restricted maximum likelihood (REML) criterion. It is used to assess the goodness of fit of the model. Lower values indicate a better fit.

Scaled Residuals: These statistics provide information about the residuals (differences between observed and predicted values) in the model:

Min: The minimum scaled residual is approximately -1.409. 1Q (First Quartile): The first quartile (25th percentile) of scaled residuals is around -0.5909. Median (50th Percentile): The median is about -0.3008. 3Q (Third Quartile): The third quartile (75th percentile) of scaled residuals is around 0.2167. Max: The maximum scaled residual is approximately 4.0113. Random Effects: This section provides information about the variance components of the random effects. In your model, you have two random effects:

"zone" with an intercept has a variance of approximately 2.857e+11 and a standard deviation of approximately 534,477. Residual variance is approximately 3.687e+12, with a standard deviation of approximately 1,920,277. Number of Observations and Groups: The number of observations in your dataset is 2,856, and the data is grouped by "zone" with 7 distinct levels.

Fixed Effects: This section provides information about the fixed effects in your model:

The estimated intercept (Intercept) is approximately 1,878,000 with a standard error of 222,300. It is highly significant with a t-value of 8.450. The estimated coefficient for "Distance" is approximately -57,480 with a standard error of 35,990. It does not appear to be statistically significant as its p-value is greater than 0.05. Significance Levels: The "Signif. codes" section indicates the significance levels of the fixed effects. In your case, the "Intercept" is highly significant (***), while "Distance" is not significant.

Correlation of Fixed Effects: The correlation value between the fixed effects is provided. In this case, it shows the correlation between the intercept and the "Distance" coefficient, which is approximately -0.229.

In summary, your linear mixed-effects model with random intercepts for “zone” includes a fixed effect for “Distance” and an intercept. The intercept is statistically significant, but the “Distance” coefficient is not. The model accounts for the hierarchical structure in your data due to the grouping by “zone.” It’s important to consider the specific context of your analysis and the research question when interpreting the results and deciding whether the model adequately captures the relationships in your data.

User

2 / 2

summary(dflmer1) Linear mixed model fit by REML. t-tests use Satterthwaite’s method [‘lmerModLmerTest’] Formula: price ~ Distance + (1 | zone) Data: df Control: lmerControl(optimizer = “Nelder_Mead”)

REML criterion at convergence: 98038.1

Scaled residuals: Min 1Q Median 3Q Max -1.1648 -0.4622 -0.2371 0.0759 11.7552

Random effects: Groups Name Variance Std.Dev. zone (Intercept) 1.021e+12 1010601 Residual 1.161e+13 3406983 Number of obs: 2979, groups: zone, 7

Fixed effects: Estimate Std. Error df t value Pr(>|t|)
(Intercept) 1.890e+06 4.147e+05 6.866e+00 4.558 0.002743 ** Distance 2.194e+05 5.928e+04 9.572e+03 3.701 0.000216 *** — Signif. codes: 0 ‘’ ***0.001 ‘’*** *0.01 ‘’* 0.05 ‘.’ 0.1 ‘ ’ 1

Correlation of Fixed Effects: (Intr) Distance -0.209 ChatGPT The output you provided is the summary of a linear mixed-effects model (LMM) fitted to your data. This LMM has a random intercept for the grouping variable “zone.” Here’s an interpretation of the key information in this summary:

Formula: The formula specifies the model you fitted. In this case, you have a fixed effect for “Distance” and a random intercept for “zone,” which is indicated by (1 | zone).

Control: This section specifies additional details about the fitting process, such as the optimization method used.

REML Criterion at Convergence: This is the value of the residual sum of squares based on the restricted maximum likelihood (REML) criterion. It is used to assess the goodness of fit of the model. Lower values indicate a better fit. In this case, the REML criterion is 98038.1.

Scaled Residuals: These statistics provide information about the residuals (differences between observed and predicted values) in the model:

Min: The minimum scaled residual is approximately -1.1648. 1Q (First Quartile): The first quartile (25th percentile) of scaled residuals is around -0.4622. Median (50th Percentile): The median is about -0.2371. 3Q (Third Quartile): The third quartile (75th percentile) of scaled residuals is around 0.0759. Max: The maximum scaled residual is approximately 11.7552. Random Effects: This section provides information about the variance components of the random effects. In your model, you have two random effects:

“zone” with an intercept has a variance of approximately 1.021e+12 and a standard deviation of approximately 1,010,601. Residual variance is approximately 1.161e+13, with a standard deviation of approximately 3,406,983. Number of Observations and Groups: The number of observations in your dataset is 2,979, and the data is grouped by “zone” with 7 distinct levels.

Fixed Effects: This section provides information about the fixed effects in your model:

The estimated intercept (Intercept) is approximately 1,890,000 with a standard error of 414,700. It is statistically significant with a t-value of 4.558 and a p-value of 0.002743. The estimated coefficient for “Distance” is approximately 219,400 with a standard error of 59,280. It is also statistically significant with a t-value of 3.701 and a p-value of 0.000216. Significance Levels: The “Signif. codes” section indicates the significance levels of the fixed effects. In your case, both the “Intercept” and “Distance” are statistically significant with different levels of significance (the “Intercept” is marked with “ ” and “Distance” with “*“).

Correlation of Fixed Effects: The correlation value between the intercept and the "Distance" coefficient is provided. In this case, it is approximately -0.209.

In summary, your linear mixed-effects model with a random intercept for "zone" includes fixed effects for "Distance." Both the intercept and the "Distance" coefficient are statistically significant. The model accounts for the hierarchical structure in your data due to the grouping by "zone." It's important to consider the specific context of your analysis and the research question when interpreting the results and deciding whether the model adequately captures the relationships in your data.

User can you give me more detail on what the numbers mean ChatGPT Certainly, I can provide more detail on the various numbers and components in the summary of your linear mixed-effects model:

Random Effects:

Groups: You have 7 unique groups or levels of the "zone" variable in your dataset. Each group represents a different zone or category for which random intercepts are estimated.

Name (Random Intercept): The "Intercept" under the "zone" group represents the random intercept term. This term accounts for the variability in the "price" variable that is specific to each zone, beyond what is explained by the fixed effects.

Variance: The estimated variance for the random intercept term under "zone" is approximately 1.021e+12. This value quantifies the extent to which the prices within each zone tend to deviate from the overall average price.

Standard Deviation (Std.Dev.): The standard deviation of the random intercept is approximately 1,010,601. It indicates the typical amount by which prices within each zone tend to vary from the average price.

Fixed Effects:

(Intercept): This represents the estimated intercept for the fixed effects. It is the predicted value of the "price" variable when "Distance" is zero.

Estimate: The estimated intercept is approximately 1,890,000. This is the predicted price when "Distance" is zero.

Std. Error: The standard error of the estimate is approximately 414,700. It measures the uncertainty or variability in the estimate.

t value: The t-value (4.558) is a measure of how many standard errors the estimate is away from zero. A higher absolute t-value indicates a more significant effect.

Pr(>|t|): The p-value (0.002743) associated with the t-value tests the null hypothesis that the effect of the intercept is zero. In this case, the small p-value suggests that the intercept is statistically significant.

Distance: This represents the estimated coefficient for the "Distance" variable. It shows how the "price" variable changes as "Distance" changes.

Estimate: The estimated coefficient for "Distance" is approximately 219,400. It represents the change in the predicted price for a one-unit change in the "Distance" variable.

Std. Error: The standard error of the estimate is approximately 59,280, indicating the uncertainty in the estimate.

t value: The t-value (3.701) is a measure of how many standard errors the estimate is away from zero. It indicates the significance of the effect of "Distance" on the price.

Pr(>|t|): The p-value (0.000216) associated with the t-value tests the null hypothesis that the effect of "Distance" is zero. The small p-value suggests that "Distance" is statistically significant.

Correlation of Fixed Effects:

The correlation of -0.209 between the intercept and the "Distance" coefficient indicates the degree of linear association between these two fixed effects. In this case, the correlation is negative, suggesting a negative relationship between the intercept and the effect of “Distance.” This means that as the intercept increases (the predicted price when “Distance” is zero), the effect of “Distance” tends to decrease.

In summary, the numbers and components in the summary of your linear mixed-effects model provide information about the model’s fixed effects, random effects, their significance, and the relationships between the variables. The model accounts for variations in “price” due to both the “Distance” predictor and the different “zone” groups. The significance levels help determine whether the effects are statistically significant in explaining the variation in housing prices.

User dflmer1 %>% fixef() %>% round(5) (Intercept) Distance 1890185.7 219352.7 ChatGPT The fixef() function in R is used to extract the fixed effects (coefficients) from a linear mixed-effects model. The output you provided shows the estimated fixed effects for your model, which include the intercept and the coefficient for the “Distance” variable. Here are the estimated fixed effects:

The estimated intercept (Intercept) is approximately 1,890,185.7. This is the predicted price when the “Distance” variable is zero.

The estimated coefficient for “Distance” is approximately 219,352.7. It represents the change in the predicted price for a one-unit change in the “Distance” variable.

These values are consistent with what was shown in the summary of your linear mixed-effects model. The intercept and “Distance” coefficient, along with their associated standard errors, t-values, and p-values, provide insights into how these variables contribute to the model’s predictions and their statistical significance.

User does that mean price will decrease by 219352 every mile away ChatGPT Yes, based on the estimated coefficient for the “Distance” variable in your linear mixed-effects model, it appears that the model predicts that the price will decrease by approximately 219,352.7 units (e.g., currency units) for each additional mile of distance. In other words, as the distance from a certain point (likely a reference point) increases by one mile, the model expects the price to decrease by the specified amount.

However, it’s important to keep in mind that this is a simplified interpretation of the model. The coefficient represents the average effect of distance on price, assuming all other factors in the model remain constant. Real-world relationships can be more complex, and other factors may also influence housing prices. Additionally, the negative effect of distance may not hold uniformly across all data points, and there may be variations within different zones, as accounted for by the random intercepts in your mixed-effects model.

You should consider the model’s assumptions, limitations, and the specific context of your data when interpreting and applying the coefficient for “Distance” to make predictions or inferences about the impact of distance on housing prices.

User

2 / 2

dflmer1 %>% se.coef() %>% extract2(1) %>% round(5) [1] 414721.1 59276.0 ChatGPT The code you provided appears to extract standard errors for the coefficients of your linear mixed-effects model. Here are the standard errors for the model’s coefficients:

The standard error for the intercept is approximately 414,721.1. The standard error for the “Distance” coefficient is approximately 59,276.0. Standard errors provide a measure of the uncertainty associated with the coefficient estimates. In this case, a larger standard error indicates greater uncertainty in the estimate. Smaller standard errors generally suggest more precise estimates.

These standard errors are consistent with what you would typically find in the summary output of a linear mixed-effects model, as shown earlier. The standard errors are used to compute t-values, which, in turn, are used to determine the statistical significance of the coefficients.

User dflmer1diag <- data.frame(residuals=resid(dflmer1), zone=df$zone, Fitted=fitted(dflmer1)) ggplot(data=dflmer1diag, aes(x=Fitted, y=residuals, col=zone))+ geom_point()+ facet_wrap(~zone)+ ggtitle(“lowest level residuals facettign by zone”) ChatGPT It looks like you’re creating a diagnostic plot for your linear mixed-effects model, with residuals plotted against the fitted values and faceted by the “zone” variable using the ggplot2 package. This is a useful way to visually inspect the model's residuals to check for any patterns or trends. The code you provided is correctly setting up the plot. Here's a breakdown of the code:

dflmer1diag <- data.frame(residuals = resid(dflmer1), zone = df$zone, Fitted = fitted(dflmer1)): This code creates a new data frame dflmer1diag that contains three columns: "residuals" (the residuals of the model), "zone" (the zone variable from your dataset), and "Fitted" (the fitted values from your model).

ggplot(data = dflmer1diag, aes(x = Fitted, y = residuals, col = zone)): This sets up a ggplot object using the data from dflmer1diag. It specifies that the x-axis should represent the fitted values, the y-axis should represent the residuals, and that points should be colored by the "zone" variable.

geom_point(): This adds a scatterplot layer to the ggplot, which plots points for each combination of fitted values and residuals.

facet_wrap(~zone): This facetting command creates separate panels for each level of the "zone" variable, so you can see the residuals for each zone separately.

ggtitle("lowest level residuals faceting by zone"): This adds a title to your plot.

When you execute this code, you will generate a series of scatterplots where each panel represents a different "zone." These panels allow you to visually inspect the distribution and patterns of residuals within each zone, which can help you assess the model's performance and identify any potential issues, such as heteroscedasticity or outliers.

User

3 / 3

what does the fitted mean ChatGPT In the context of linear regression models, "fitted" or "fitted values" refer to the predicted values of the dependent variable (in your case, "price") based on the model's coefficients and the values of the independent variables (in your case, "Distance" and potentially others).

When you fit a linear regression model, the goal is to find a mathematical equation that best represents the relationship between the independent variables and the dependent variable. The fitted values are the values of the dependent variable that the model predicts for each set of independent variable values.

For example, in your linear mixed-effects model, you have an intercept and a coefficient for "Distance." The fitted values for each observation in your dataset are obtained by applying these coefficients to the values of "Distance" for that observation.

In a scatterplot of fitted values against actual observed values, you can assess how well the model approximates the actual data. Ideally, the points should cluster closely around a diagonal line, indicating a good fit. If there are systematic deviations from the diagonal line, it may suggest that the model is not capturing the relationship between the independent and dependent variables accurately. Residuals, the differences between the observed and fitted values, are used to assess the model's accuracy and identify any patterns or outliers in the data.

In summary, "fitted values" are the predicted values of the dependent variable produced by a regression model based on the model's coefficients and the values of the independent variables. They are used to assess how well the model explains the variation in the observed data.

User so teh fitted just means price ChatGPT Yes, in the context of your linear mixed-effects model, the "Fitted" values represent the predicted or estimated values of the "price" variable. These values are generated by your model based on the observed values of the independent variables, such as "Distance" and the model's coefficients (including the intercept and the coefficient for "Distance"). The "Fitted" values are the model's predictions for the "price" variable, and they are sometimes referred to as the "fitted values" because they are the values that the model fits to the observed data.

User

2 / 2

what does a random effect show ChatGPT Random effects in a linear mixed-effects model capture variability in the dependent variable (response) that is not explained by the fixed effects but is associated with the grouping or hierarchical structure of the data. In your case, you have specified a random intercept for the "zone" variable in your linear mixed-effects model. Here's what a random intercept shows:

Group-Level Variation: A random intercept accounts for variation in the response variable (in your case, "price") at the group level. In your model, each "zone" represents a different group. The random intercept allows the model to estimate how the average "price" varies from one zone to another.

Group-Specific Differences: Random intercepts allow each group (zone) to have its own unique average "price." This means that your model recognizes that there may be systematic differences in housing prices between different zones that are not accounted for by the fixed effects.

Partial Pooling: Random intercepts strike a balance between fully pooling the data (assuming a single common average for all zones) and not pooling at all (treating each zone entirely independently). Partial pooling means that information is shared between the groups, which can be especially useful when there are few observations within each group.

Variability Estimation: Random intercepts estimate the variance of "price" between different zones. The variance component reflects how much the average "price" can vary from one zone to another. A larger estimated variance implies more substantial variation in housing prices between zones.

Improving Model Fit: By including random intercepts, you account for the hierarchical structure of your data, reducing the risk of underestimating or overestimating the variability in housing prices and improving the model's fit to the data.

In summary, a random intercept in a linear mixed-effects model acknowledges that there are systematic differences in the response variable among different groups (in your case, zones) and aims to estimate the variability in the response variable between these groups. It allows the model to capture and account for group-specific patterns and differences that may not be explained by the fixed effects.