In-Class Analysis 2

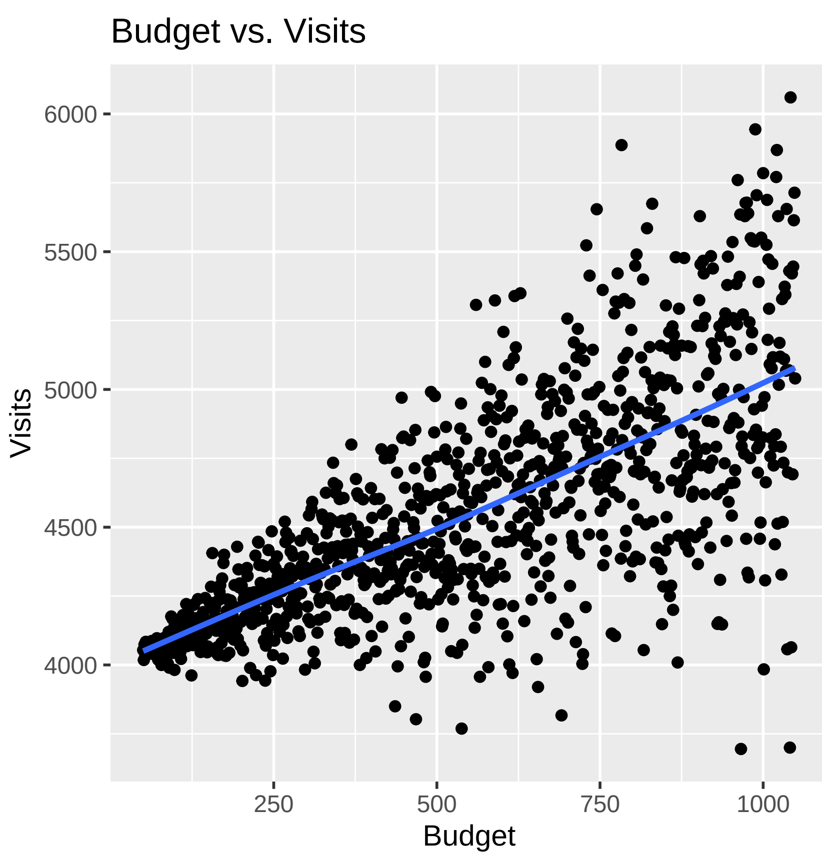
Jillian Warburton

2023-02-03

Exploratory Data Analysis

1. Draw a scatterplot of `Visits` by `Budget`. Add a smooth line to gauge how linear the relationship is.

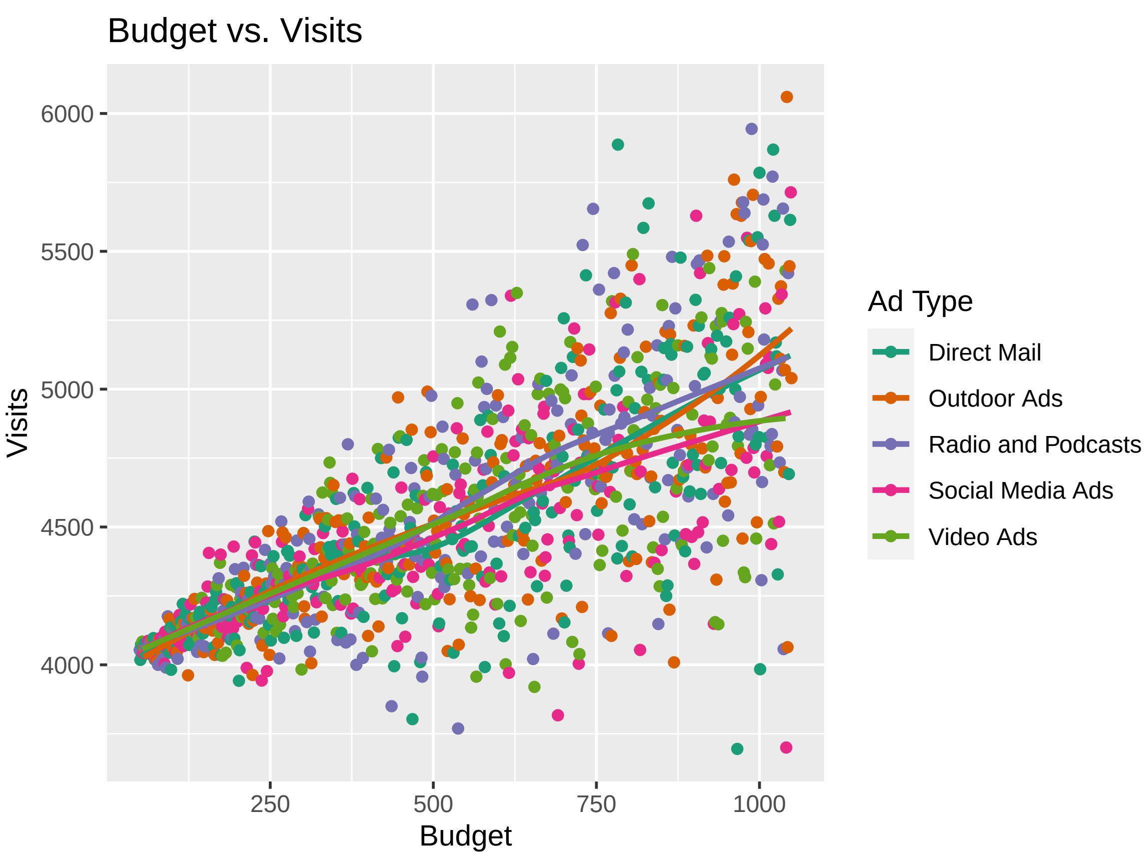
```r
#scatterplot of Budget vs Visits with trend line and axis labels
ggplot(data = data, mapping = aes(x = Budget, y = Visits)) +
  geom_point() +
  theme(aspect.ratio = 1) +
  ggtitle("Budget vs. Visits") +
  xlab("Budget") +
  ylab("Visits") +
  geom_smooth(se=FALSE)
```



The relationship between `Budget` and `Visits` appear very linear.

2. Draw a scatterplot of `Visits` by `Budget` colored by `AdType`. Add a smooth line to gauge how linear the relationship is and if there is a possible interaction between `Budget` and `AdType`.

```
#scatterplot of Budget vs Visits, colored by AdType with trend line and axis labels
ggplot(data = data, mapping = aes(x = Budget, y = Visits, color=AdType)) +
  geom_point() +
  theme(aspect.ratio = 1) +
  ggtitle("Budget vs. Visits") +
  xlab("Budget") +
  ylab("Visits") +
  geom_smooth(se=FALSE) +
  scale_color_brewer(palette = "Dark2", name = 'Ad Type')
```



The relationship between `Budget` and `Visits`, when accounting for possible interactions between `Budget` and `AdType`, still appears linear. There does not appear to be any interactions between `Budget` and `AdType`.

Analysis with a MLR

1. Fit a MLR model for Visits using `Budget` and `AdType` as explanatory variables. Build a 95% confidence interval for the effect of these variables on `Visits` (this is the first attempt at answering research question #1).

```
ad.lm <- lm(formula = Visits~.-Company, data = data)
ad.ci <- confint(ad.lm, level = 0.95)
ad.ci
```

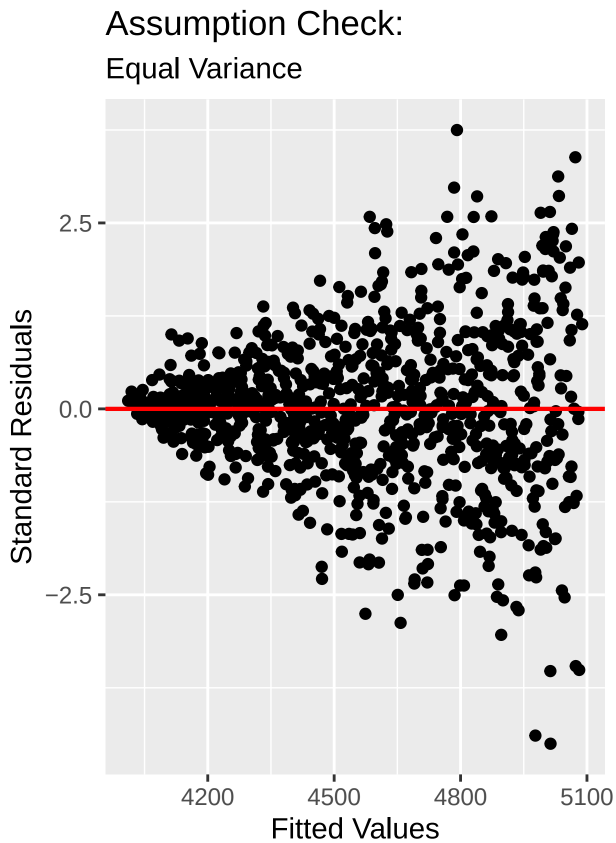
```
##                                  2.5 %      97.5 %
## (Intercept)              3944.2268499 4046.646797
## Budget                      0.9535093    1.079901
## AdTypeOutdoor Ads         -39.2009793   74.446248
## AdTypeRadio and Podcasts  -25.1293745   88.698130
## AdTypeSocial Media Ads    -97.9132287   17.337885
## AdTypeVideo Ads           -66.7589827   46.023669
```

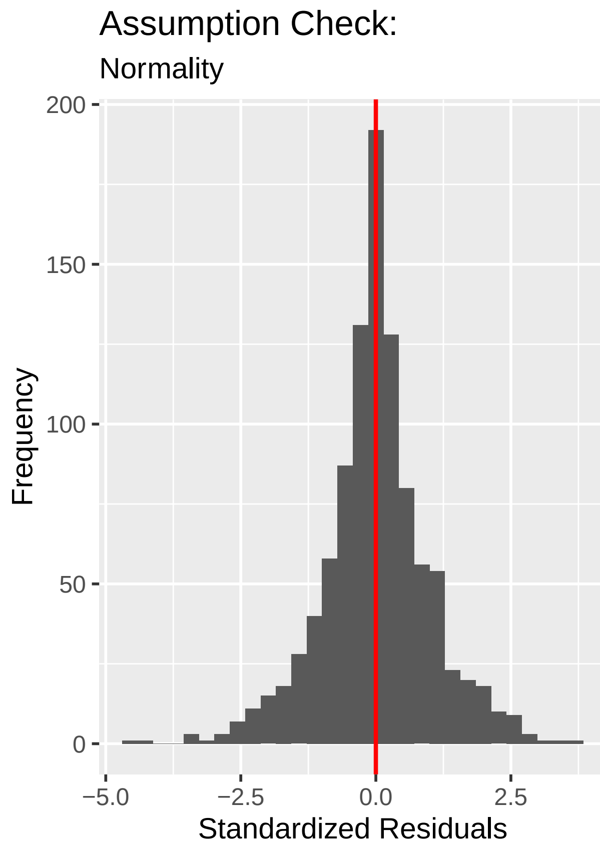
2. Show that the assumptions of the MLR model are not met (and hence statistical inference using an MLR model are not valid) by drawing a scatterplot of fitted vs. standardized residuals and a histogram of the standardized residuals.

```r
#check equal variance assumption with fit val vs std res scatterplot
ad.resids_fit <- ggplot(data, aes(x=ad.lm$fitted.values, y=stdres(ad.lm))) +
  geom_point() +
  xlab('Fitted Values') +
  ylab('Standard Residuals') +
  ggtitle('Assumption Check:', subtitle ='Equal Variance') +
  geom_hline(yintercept = 0, col = "red", lwd = 0.75)
```

```
## Warning: Using ‘size‘ aesthetic for lines was deprecated in ggplot2 3.4.0.
## i Please use ‘linewidth‘ instead.
```

```r
#check normality assumption with std res histogram
stdres.freq <- ggplot() +
  geom_histogram(mapping=aes(x=stdres(ad.lm))) +
  xlab('Standardized Residuals') +
  ylab('Frequency') +
  ggtitle('Assumption Check:', subtitle = 'Normality') +
  geom_vline(xintercept = 0, col = "red", lwd = 0.75)

suppressMessages(grid.arrange(ad.resids_fit, stdres.freq, nrow=1))
```

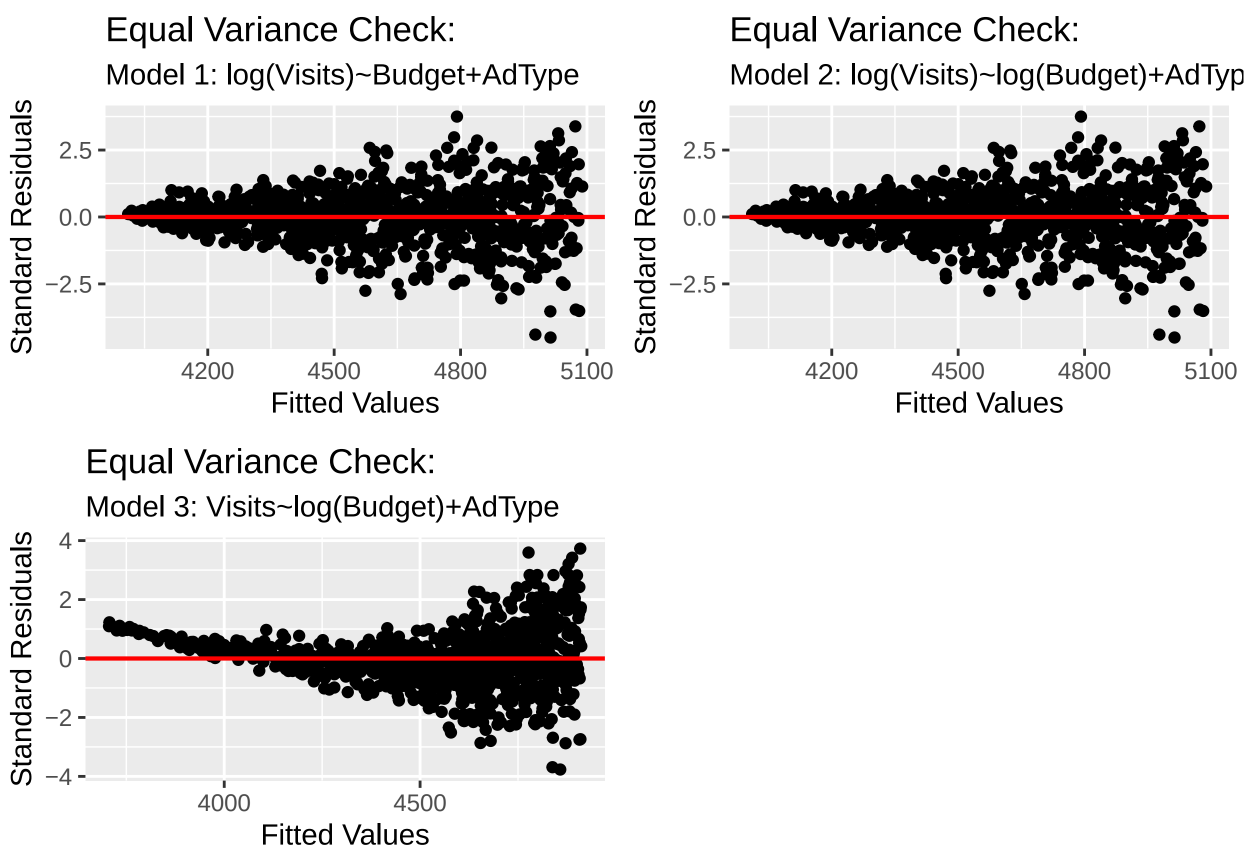




3. Try 2 or 3 transformations of `Visits` and/or `Budget` and show that these transformations are not going to fix the assumptions of the MLR model.

```
ad.lm1 <- lm(I(log(Visits))~Budget+AdType, data)
#check equal variance assumption with fit val vs std res scatterplot
ad.resids_fit1 <- ggplot(data, aes(x=ad.lm$fitted.values, y=stdres(ad.lm))) +
  geom_point() +
  xlab('Fitted Values') +
  ylab('Standard Residuals') +
  ggtitle('Equal Variance Check:', subtitle ='Model 1: log(Visits)~Budget+AdType') +
  geom_hline(yintercept = 0, col = "red", lwd = 0.75)

ad.lm2 <- lm(I(log(Visits))~log(Budget)+AdType, data)
#check equal variance assumption with fit val vs std res scatterplot
ad.resids_fit2 <- ggplot(data, aes(x=ad.lm$fitted.values, y=stdres(ad.lm))) +
  geom_point() +
  xlab('Fitted Values') +
  ylab('Standard Residuals') +
  ggtitle('Equal Variance Check:', subtitle ='Model 2: log(Visits)~log(Budget)+AdType') +
  geom_hline(yintercept = 0, col = "red", lwd = 0.75)

ad.lm3 <- lm(Visits~log(Budget)+AdType, data)
#check equal variance assumption with fit val vs std res scatterplot
ad.resids_fit3 <- ggplot(data, aes(x=ad.lm3$fitted.values, y=stdres(ad.lm3))) +
  geom_point() +
```

```
  xlab('Fitted Values') +
  ylab('Standard Residuals') +
  ggtitle('Equal Variance Check:', subtitle ='Model 3: Visits~log(Budget)+AdType') +
  geom_hline(yintercept = 0, col = "red", lwd = 0.75)

suppressMessages(grid.arrange(ad.resids_fit1, ad.resids_fit2, ad.resids_fit3, nrow = 2))
```



I tried taking the log of `Visits`, `Budget`, and both `Visits` and `Budget`, but it did not fix the equal variance assumption of the MLR model.

Fitting a Linear Model with Heteroskedasticity

1. Fit a heteroskedastic MLR model to the website data where `Visits` is the response, `Budget` and `AdType` are the explanatory variables and we `Budget` is the covariate in an exponential variance function. Identify the estimates of $\hat{\beta}$, $\hat{\theta}$ and s.

```
adj.ad.lm <- gls(model= Visits~Budget+AdType, data= data,
                 weights=varExp(form=~Budget), method="ML")
print("Beta Estimates:")
```

```
## [1] "Beta Estimates:"
```

```
adj.ad.lm $coefficients
```

```
##              (Intercept)                    Budget        AdTypeOutdoor Ads
##             3994.3229394                 1.0051835                8.1031987
## AdTypeRadio and Podcasts   AdTypeSocial Media Ads           AdTypeVideo Ads
##               10.5905337                 0.3641742               10.5673467
```

```r
print("Theta Estimate:")
```

```
## [1] "Theta Estimate:"
```

```r
coef(adj.ad.lm $modelStruct, unconstrained=FALSE)
```

```
## varStruct.expon
##     0.002014862
```

```r
print("Variance Estimate")
```

```
## [1] "Variance Estimate"
```

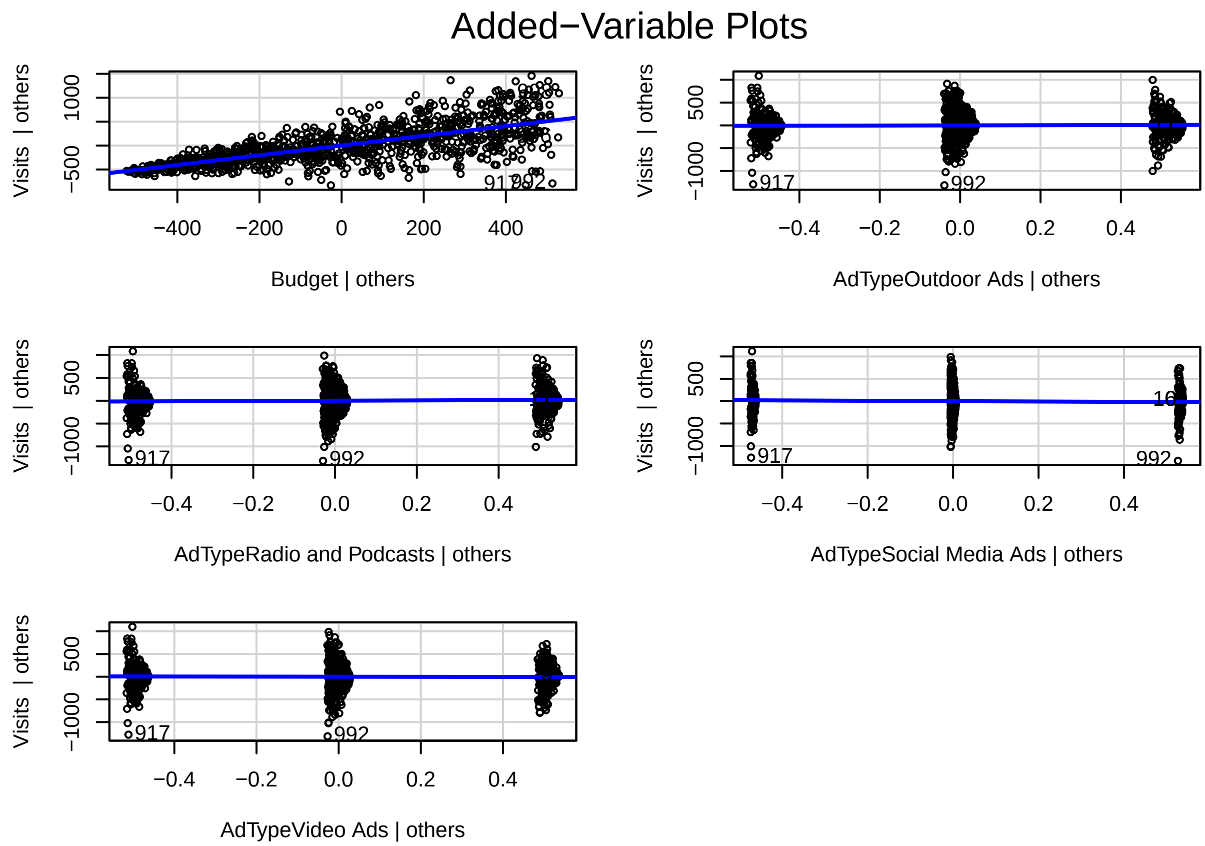
```r
adj.ad.lm $sigma
```

```
## [1] 76.40797
```

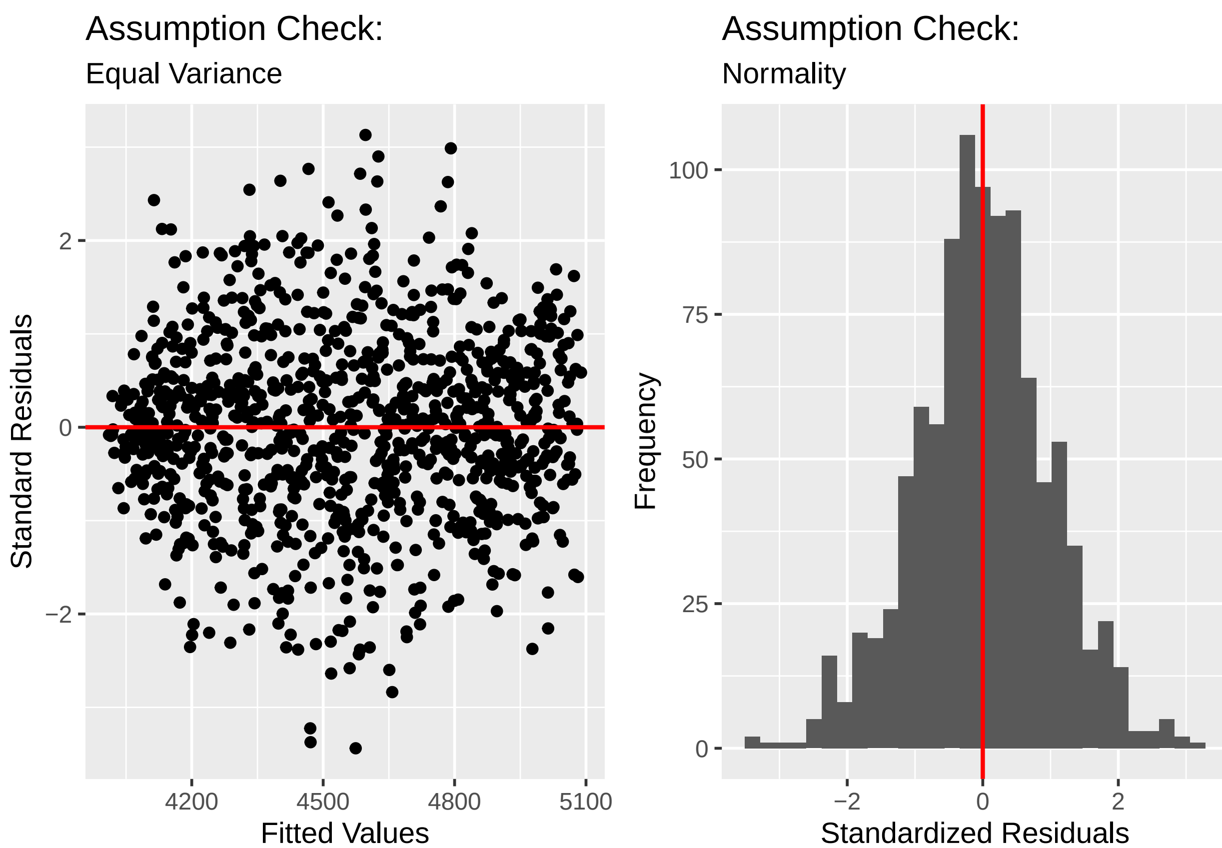
Validating your Heteroskedastic MLR Model

1. Check the L-I-N-E assumptions using the standardized residuals from your heterogeneous MLR fit in the previous subsection.

```r
#Linear assumption
avPlots(ad.lm, ask = FALSE)
```




```r
#Equal Variance assumption
resids = resid(object=adj.ad.lm, type="pearson")
ad.resids_fit2 <- ggplot(data, aes(x=ad.lm$fitted.values, y=resids)) +
  geom_point() +
  xlab('Fitted Values') +
  ylab('Standard Residuals') +
  ggtitle('Assumption Check:', subtitle ='Equal Variance') +
  geom_hline(yintercept = 0, col = "red", lwd = 0.75)
#Normality Assumption
stdres.freq2 <- ggplot() +
  geom_histogram(mapping=aes(x=resids)) +
  xlab('Standardized Residuals') +
  ylab('Frequency') +
  ggtitle('Assumption Check:', subtitle = 'Normality') +
  geom_vline(xintercept = 0, col = "red", lwd = 0.75)

suppressMessages(grid.arrange(ad.resids_fit2, stdres.freq2, nrow=1))
```



We can safely assume that the visits of one website do not affect the visits of another website. The independence assumption is fulfilled. The graphs above show that the linearity, equal variance, and normality assumptions are fulfilled.

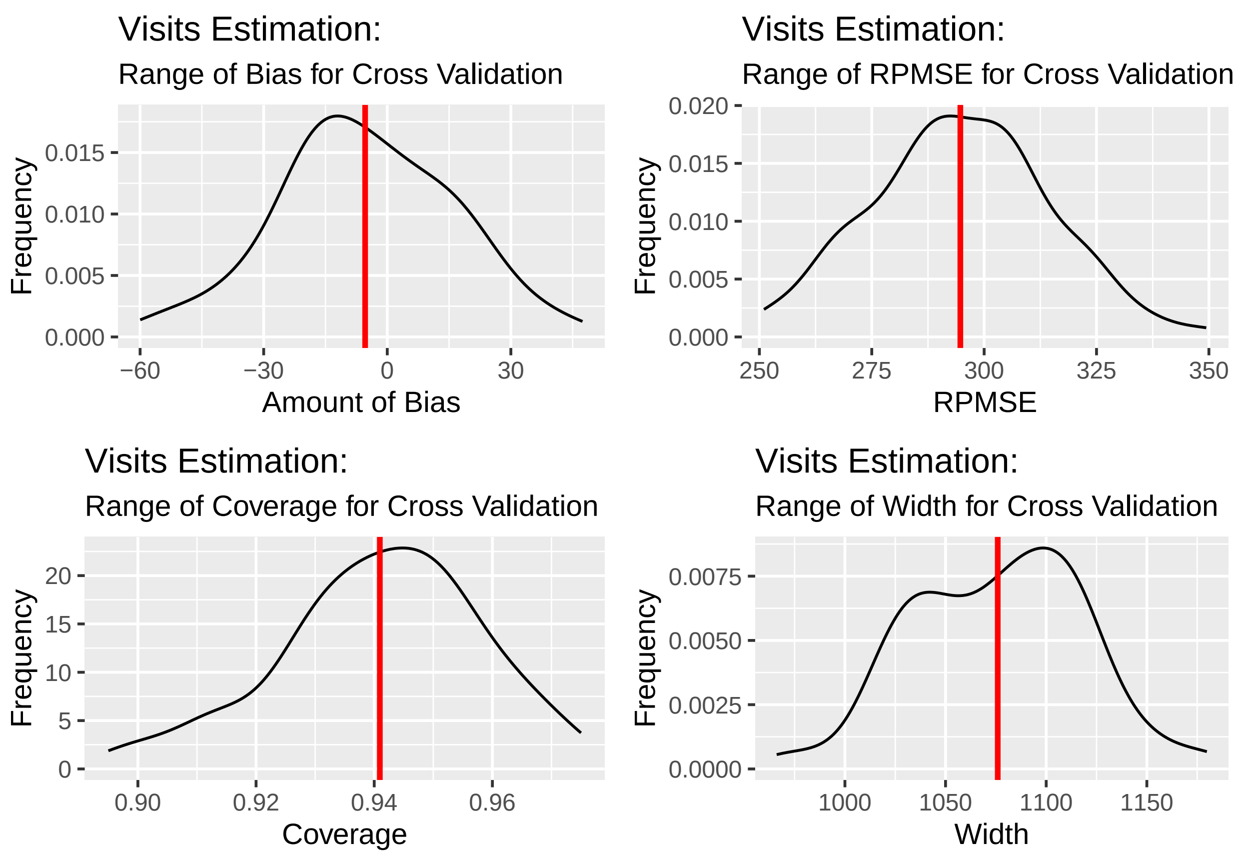
```
source("predictgls.R")
```

2. Modify the cross-validation code from the birth weight analysis to run a cross-validation of your heterogeneous MLR using the `predictgls()` function. Report the bias, RPMSE, coverage and width of prediction intervals.

```
set.seed(29) #for reproducibility
n.cv <- 100 #Number of CV studies to run
n.test <- 200 #Number of observations in a test set
# n.test = 200 is about 20% of 1000
rpmse <- rep(x=NA, times=n.cv)
bias <- rep(x=NA, times=n.cv)
wid <- rep(x=NA, times=n.cv)
cvg <- rep(x=NA, times=n.cv)
for(cv in 1:n.cv){
  ## Select test observations
  test.obs <- sample(x=1:nrow(data), size=n.test)

  ## Split into test and training sets
  test.set <- data[test.obs,]
  train.set <- data[-test.obs,]
```

```r
  ## Fit a lm() using the training data
  train.lm <- gls(model=Visits~Budget+AdType, data=train.set,
                 weights=varExp(form=~Budget), method="ML")

  ## Generate predictions for the test set
  my.preds <- predictgls(train.lm, newdframe = test.set)

  ## Calculate bias
  bias[cv] <- mean(my.preds[,'Prediction']-test.set[['Visits']])

  ## Calculate RPMSE
  rpmse[cv] <- (test.set[['Visits']]-my.preds[,'Prediction'])^2 %>% mean() %>% sqrt()

  ## Calculate Coverage
  cvg[cv] <- ((test.set[['Visits']] > my.preds[,'lwr']) &
               (test.set[['Visits']] < my.preds[,'upr'])) %>% mean()

  ## Calculate Width
  wid[cv] <- (my.preds[,'upr'] - my.preds[,'lwr']) %>% mean()

}

CV.bias <- ggplot() +
  geom_density(mapping=aes(x=bias)) +
  xlab('Amount of Bias') +
  ylab('Frequency') +
  ggtitle('Visits Estimation:',
          subtitle = 'Range of Bias for Cross Validation') +
  geom_vline(xintercept = mean(bias), col = "red", lwd = 1)

CV.RPMSE <- ggplot() +
  geom_density(mapping=aes(x=rpmse)) +
  xlab('RPMSE') +
  ylab('Frequency') +
  ggtitle('Visits Estimation:',
          subtitle = 'Range of RPMSE for Cross Validation') +
  geom_vline(xintercept = mean(rpmse), col = "red", lwd = 1)

CV.coverage <- ggplot() +
  geom_density(mapping=aes(x=cvg)) +
  xlab('Coverage') +
  ylab('Frequency') +
  ggtitle('Visits Estimation:',
          subtitle = 'Range of Coverage for Cross Validation') +
  geom_vline(xintercept = mean(cvg), col = "red", lwd = 1)

CV.width <- ggplot() +
  geom_density(mapping=aes(x=wid)) +
  xlab('Width') +
  ylab('Frequency') +
  ggtitle('Visits Estimation:',
          subtitle = 'Range of Width for Cross Validation') +
  geom_vline(xintercept = mean(wid), col = "red", lwd = 1)
```

```r
suppressMessages(grid.arrange(CV.bias, CV.RPMSE, CV.coverage, CV.width, nrow=2))
```



The red lines on each graph show the means of the reported metric.

3. For each company in your dataset and only the social media ad types, construct a 99% prediction interval for `Visits`.

```r
newdf <- data %>% filter(AdType=="Social Media Ads")
dataPreds <- predictgls(glsobj=adj.ad.lm, level=0.99, newdframe=newdf)
# ggplot() +
#   geom_point(data=dataPreds %>% filter(AdType=="Social Media Ads"),
#              mapping=aes(x=Budget, y=Visits)) + #Scatterplot
#   geom_line(data=dataPreds %>% filter(AdType=="Social Media Ads"),
#             mapping=aes(x=Budget, y=Prediction)) + #Prediction Line
#   geom_line(data=dataPreds %>% filter(AdType=="Social Media Ads"),
#             mapping=aes(x=Budget, y=lwr),
#             color="red", linetype="dashed") + #lwr bound
#   geom_line(data=dataPreds %>% filter(AdType=="Social Media Ads"),
#             mapping=aes(x=Budget, y=upr),
#             color="red", linetype="dashed") #Upper bound
print(dataPreds)
```

```
##      Company Budget Visits           AdType Prediction   SE.pred      lwr
## 1001       6     55   4043 Social Media Ads   4049.972  86.38129 3827.041
## 1002      10     59   4046 Social Media Ads   4053.993  87.05736 3829.317
```

```
## 1003      13     62   4050 Social Media Ads   4057.008  87.56827 3831.014
## 1004      14     63   4087 Social Media Ads   4058.014  87.73932 3831.577
## 1005      15     64   4057 Social Media Ads   4059.019  87.91073 3832.140
## 1006      18     67   4038 Social Media Ads   4062.034  88.42719 3833.823
## 1007      27     76   4013 Social Media Ads   4071.081  89.99678 3838.819
## 1008      33     82   4098 Social Media Ads   4077.112  91.06019 3842.105
## 1009      39     88   4004 Social Media Ads   4083.143  92.13737 3845.357
## 1010      40     89   4120 Social Media Ads   4084.148  92.31825 3845.895
## 1011      43     92   4064 Social Media Ads   4087.164  92.86321 3847.504
## 1012      56    105   4045 Social Media Ads   4100.231  95.26547 3854.372
## 1013      61    110   4083 Social Media Ads   4105.257  96.20726 3856.967
## 1014      62    111   4181 Social Media Ads   4106.262  96.39682 3857.483
## 1015      65    114   4118 Social Media Ads   4109.278  96.96793 3859.025
## 1016      68    117   4068 Social Media Ads   4112.294  97.54268 3860.557
## 1017      72    121   4109 Social Media Ads   4116.314  98.31471 3862.585
## 1018      79    128   4220 Social Media Ads   4123.351  99.68160 3866.094
## 1019      88    137   4114 Social Media Ads   4132.397 101.46896 3870.528
## 1020      95    144   4122 Social Media Ads   4139.434 102.88274 3873.915
## 1021     103    152   4226 Social Media Ads   4147.475 104.52415 3877.721
## 1022     104    153   4153 Social Media Ads   4148.480 104.73128 3878.191
## 1023     105    154   4284 Social Media Ads   4149.485 104.93883 3878.661
## 1024     107    156   4406 Social Media Ads   4151.496 105.35525 3879.597
## 1025     109    158   4225 Social Media Ads   4153.506 105.77342 3880.528
## 1026     125    174   4400 Social Media Ads   4169.589 109.18247 3887.813
## 1027     127    176   4138 Social Media Ads   4171.599 109.61667 3888.702
## 1028     130    179   4238 Social Media Ads   4174.615 110.27136 3890.028
## 1029     131    180   4172 Social Media Ads   4175.620 110.49051 3890.468
## 1030     136    185   4101 Social Media Ads   4180.646 111.59309 3892.648
## 1031     137    186   4160 Social Media Ads   4181.651 111.81499 3893.081
## 1032     139    188   4111 Social Media Ads   4183.662 112.26017 3893.942
## 1033     144    193   4132 Social Media Ads   4188.688 113.38125 3896.075
## 1034     145    194   4429 Social Media Ads   4189.693 113.60687 3896.498
## 1035     151    200   4157 Social Media Ads   4195.724 114.97045 3899.010
## 1036     159    208   4206 Social Media Ads   4203.765 116.81516 3902.291
## 1037     163    212   4253 Social Media Ads   4207.786 117.74903 3903.901
## 1038     165    214   3989 Social Media Ads   4209.796 118.21888 3904.699
## 1039     172    221   4205 Social Media Ads   4216.833 119.87873 3907.452
## 1040     173    222   4397 Social Media Ads   4217.838 120.11781 3907.840
## 1041     178    227   4444 Social Media Ads   4222.864 121.32068 3909.761
## 1042     183    232   4362 Social Media Ads   4227.890 122.53603 3911.651
## 1043     185    234   4266 Social Media Ads   4229.900 123.02569 3912.397
## 1044     188    237   3943 Social Media Ads   4232.916 123.76397 3913.507
## 1045     190    239   4203 Social Media Ads   4234.926 124.25870 3914.241
## 1046     196    245   3977 Social Media Ads   4240.957 125.75518 3916.410
## 1047     211    260   4295 Social Media Ads   4256.035 129.57817 3921.621
## 1048     212    261   4293 Social Media Ads   4257.040 129.83725 3921.958
## 1049     214    263   4268 Social Media Ads   4259.050 130.35701 3922.627
## 1050     219    268   4446 Social Media Ads   4264.076 131.66578 3924.275
## 1051     221    270   4176 Social Media Ads   4266.087 132.19306 3924.925
## 1052     224    273   4211 Social Media Ads   4269.102 132.98804 3925.889
## 1053     230    279   4322 Social Media Ads   4275.133 134.59275 3927.778
## 1054     234    283   4249 Social Media Ads   4279.154 135.67356 3929.010
## 1055     246    295   4393 Social Media Ads   4291.216 138.96961 3932.566
## 1056     259    308   4566 Social Media Ads   4304.284 142.63277 3936.179
```

```
## 1057      267     316    4293 Social Media Ads   4312.325 144.93584 3938.277
## 1058      269     318    4420 Social Media Ads   4314.335 145.51751 3938.786
## 1059      272     321    4227 Social Media Ads   4317.351 146.39447 3939.538
## 1060      282     331    4478 Social Media Ads   4327.403 149.35672 3941.945
## 1061      309     358    4218 Social Media Ads   4354.543 157.66277 3947.649
## 1062      311     360    4436 Social Media Ads   4356.553 158.29637 3948.024
## 1063      312     361    4484 Social Media Ads   4357.558 158.61414 3948.209
## 1064      325     374    4186 Social Media Ads   4370.626 162.80444 3950.463
## 1065      327     376    4675 Social Media Ads   4372.636 163.45898 3950.784
## 1066      328     377    4204 Social Media Ads   4373.641 163.78724 3950.942
## 1067      338     387    4601 Social Media Ads   4383.693 167.10684 3952.426
## 1068      341     390    4340 Social Media Ads   4386.709 168.11592 3952.838
## 1069      348     397    4401 Social Media Ads   4393.745 170.49448 3953.735
## 1070      361     410    4417 Social Media Ads   4406.812 175.00238 3955.169
## 1071      369     418    4316 Social Media Ads   4414.854 177.83609 3955.897
## 1072      388     437    4267 Social Media Ads   4433.952 184.75307 3957.144
## 1073      389     438    4272 Social Media Ads   4434.957 185.12455 3957.191
## 1074      393     442    4275 Social Media Ads   4438.978 186.61799 3957.357
## 1075      394     443    4422 Social Media Ads   4439.983 186.99324 3957.394
## 1076      396     445    4068 Social Media Ads   4441.994 187.74603 3957.462
## 1077      402     451    4643 Social Media Ads   4448.025 190.02276 3957.617
## 1078      405     454    4420 Social Media Ads   4451.040 191.17153 3957.668
## 1079      407     456    4365 Social Media Ads   4453.051 191.94126 3957.692
## 1080      408     457    4102 Social Media Ads   4454.056 192.32729 3957.701
## 1081      413     462    4452 Social Media Ads   4459.082 194.26921 3957.715
## 1082      420     469    4319 Social Media Ads   4466.118 197.02110 3957.649
## 1083      421     470    4462 Social Media Ads   4467.123 197.41742 3957.632
## 1084      425     474    4223 Social Media Ads   4471.144 199.01072 3957.540
## 1085      432     481    4356 Social Media Ads   4478.180 201.83019 3957.300
## 1086      438     487    4599 Social Media Ads   4484.211 204.27882 3957.012
## 1087      444     493    4364 Social Media Ads   4490.243 206.75732 3956.646
## 1088      451     500    4756 Social Media Ads   4497.279 209.68713 3956.122
## 1089      458     507    4257 Social Media Ads   4504.315 212.65866 3955.489
## 1090      459     508    4140 Social Media Ads   4505.320 213.08661 3955.390
## 1091      461     510    4572 Social Media Ads   4507.331 213.94509 3955.185
## 1092      475     524    4549 Social Media Ads   4521.403 220.05255 3953.495
## 1093      479     528    4466 Social Media Ads   4525.424 221.82948 3952.930
## 1094      487     536    4858 Social Media Ads   4533.465 225.42668 3951.688
## 1095      491     540    4623 Social Media Ads   4537.486 227.24717 3951.010
## 1096      493     542    4654 Social Media Ads   4539.497 228.16294 3950.657
## 1097      495     544    4424 Social Media Ads   4541.507 229.08242 3950.295
## 1098      499     548    4438 Social Media Ads   4545.528 230.93254 3949.541
## 1099      501     550    4592 Social Media Ads   4547.538 231.86322 3949.149
## 1100      502     551    4431 Social Media Ads   4548.543 232.32997 3948.950
## 1101      528     577    4708 Social Media Ads   4574.678 244.80190 3942.897
## 1102      531     580    4302 Social Media Ads   4577.694 246.28351 3942.089
## 1103      534     583    4846 Social Media Ads   4580.709 247.77412 3941.257
## 1104      536     585    4504 Social Media Ads   4582.719 248.77287 3940.690
## 1105      537     586    4335 Social Media Ads   4583.725 249.27377 3940.403
## 1106      546     595    4219 Social Media Ads   4592.771 253.82754 3937.697
## 1107      555     604    4321 Social Media Ads   4601.818 258.46470 3934.776
## 1108      566     615    4922 Social Media Ads   4612.875 264.24781 3930.908
## 1109      567     616    3971 Social Media Ads   4613.880 264.77993 3930.540
## 1110      570     619    5339 Social Media Ads   4616.896 266.38276 3929.419
```

```
## 1111      574     623    4760 Social Media Ads   4620.916 268.53499 3927.885
## 1112      577     626    4811 Social Media Ads   4623.932 270.16059 3926.706
## 1113      580     629    4609 Social Media Ads   4626.948 271.79606 3925.500
## 1114      581     630    5036 Social Media Ads   4627.953 272.34341 3925.093
## 1115      587     636    4855 Social Media Ads   4633.984 275.65082 3922.588
## 1116      597     646    4822 Social Media Ads   4644.036 281.25282 3918.183
## 1117      600     649    4336 Social Media Ads   4647.051 282.95555 3916.804
## 1118      613     662    4711 Social Media Ads   4660.119 290.45420 3910.519
## 1119      618     667    4585 Social Media Ads   4665.144 293.39101 3907.966
## 1120      620     669    4911 Social Media Ads   4667.155 294.57405 3906.923
## 1121      621     670    4936 Social Media Ads   4668.160 295.16735 3906.397
## 1122      622     671    4323 Social Media Ads   4669.165 295.76186 3905.868
## 1123      623     672    4390 Social Media Ads   4670.170 296.35756 3905.335
## 1124      626     675    4455 Social Media Ads   4673.186 298.15189 3903.720
## 1125      629     678    4652 Social Media Ads   4676.202 299.95710 3902.077
## 1126      636     685    4702 Social Media Ads   4683.238 304.21193 3898.132
## 1127      638     687    4744 Social Media Ads   4685.248 305.43866 3896.977
## 1128      642     691    3817 Social Media Ads   4689.269 307.90701 3894.627
## 1129      645     694    4569 Social Media Ads   4692.284 309.77136 3892.831
## 1130      656     705    4650 Social Media Ads   4703.341 316.70455 3885.995
## 1131      659     708    4448 Social Media Ads   4706.357 318.62225 3884.062
## 1132      666     715    4854 Social Media Ads   4713.393 323.14219 3879.433
## 1133      667     716    5220 Social Media Ads   4714.398 323.79312 3878.758
## 1134      670     719    4713 Social Media Ads   4717.414 325.75378 3876.714
## 1135      671     720    4543 Social Media Ads   4718.419 326.40997 3876.025
## 1136      674     723    4004 Social Media Ads   4721.435 328.38650 3873.940
## 1137      676     725    4733 Social Media Ads   4723.445 329.71084 3872.533
## 1138      682     731    4982 Social Media Ads   4729.476 333.71602 3868.227
## 1139      686     735    4749 Social Media Ads   4733.497 336.41316 3865.287
## 1140      689     738    4982 Social Media Ads   4736.513 338.45033 3863.045
## 1141      690     739    5144 Social Media Ads   4737.518 339.13213 3862.291
## 1142      703     752    4648 Social Media Ads   4750.585 348.12161 3852.158
## 1143      704     753    4472 Social Media Ads   4751.590 348.82291 3851.354
## 1144      714     763    4721 Social Media Ads   4761.642 355.91414 3843.104
## 1145      722     771    4627 Social Media Ads   4769.684 361.69089 3836.237
## 1146      730     779    5315 Social Media Ads   4777.725 367.56148 3829.128
## 1147      735     784    4816 Social Media Ads   4782.751 371.27893 3824.560
## 1148      742     791    4936 Social Media Ads   4789.787 376.54666 3818.001
## 1149      747     796    4322 Social Media Ads   4794.813 380.35506 3813.199
## 1150      767     816    5399 Social Media Ads   4814.917 395.97810 3792.983
## 1151      768     817    4054 Social Media Ads   4815.922 396.77590 3791.929
## 1152      769     818    4701 Social Media Ads   4816.927 397.57530 3790.871
## 1153      790     839    4370 Social Media Ads   4838.036 414.74018 3767.681
## 1154      801     850    4416 Social Media Ads   4849.093 424.02530 3754.775
## 1155      823     872    4629 Social Media Ads   4871.207 443.22426 3727.341
## 1156      838     887    4474 Social Media Ads   4886.285 456.81087 3707.354
## 1157      843     892    4726 Social Media Ads   4891.311 461.43173 3700.455
## 1158      847     896    4464 Social Media Ads   4895.332 465.16207 3694.848
## 1159      851     900    4366 Social Media Ads   4899.352 468.92259 3689.164
## 1160      854     903    5629 Social Media Ads   4902.368 471.76293 3684.849
## 1161      857     906    4481 Social Media Ads   4905.383 474.62049 3680.490
## 1162      860     909    5421 Social Media Ads   4908.399 477.49537 3676.086
## 1163      864     913    4517 Social Media Ads   4912.420 481.35566 3670.144
## 1164      866     915    4886 Social Media Ads   4914.430 483.29750 3667.143
```

```
## 1165     869    918   4729 Social Media Ads   4917.446 486.22497 3662.604
## 1166     872    921   5167 Social Media Ads   4920.461 489.17019 3658.018
## 1167     875    924   4882 Social Media Ads   4923.477 492.13325 3653.387
## 1168     881    930   4149 Social Media Ads   4929.508 498.11338 3643.984
## 1169     889    938   4638 Social Media Ads   4937.549 506.20013 3631.156
## 1170     899    948   4860 Social Media Ads   4947.601 516.49357 3614.642
## 1171     908    957   4707 Social Media Ads   4956.648 525.93662 3599.319
## 1172     911    960   5236 Social Media Ads   4959.663 529.12254 3594.112
## 1173     920    969   5272 Social Media Ads   4968.710 538.79663 3578.192
## 1174     931    980   4751 Social Media Ads   4979.767 550.86120 3558.113
## 1175     932    981   5549 Social Media Ads   4980.772 551.97129 3556.253
## 1176     942    991   4787 Social Media Ads   4990.824 563.19608 3537.336
## 1177     943    992   4698 Social Media Ads   4991.829 564.33105 3535.412
## 1178     960   1009   5293 Social Media Ads   5008.917 583.97941 3501.792
## 1179     961   1010   5091 Social Media Ads   5009.922 585.15629 3499.760
## 1180     962   1011   4757 Social Media Ads   5010.928 586.33553 3497.722
## 1181     964   1013   5077 Social Media Ads   5012.938 588.70117 3493.627
## 1182     966   1015   5117 Social Media Ads   5014.948 591.07635 3489.508
## 1183     969   1018   4438 Social Media Ads   5017.964 594.65712 3483.282
## 1184     981   1030   4519 Social Media Ads   5030.026 609.19855 3457.816
## 1185     985   1034   5344 Social Media Ads   5034.047 614.12434 3449.124
## 1186     992   1041   3700 Social Media Ads   5041.083 622.84056 3433.666
## 1187     999   1048   5714 Social Media Ads   5048.119 631.68056 3417.888
##           upr
## 1001 4272.904
## 1002 4278.669
## 1003 4283.003
## 1004 4284.450
## 1005 4285.897
## 1006 4290.246
## 1007 4303.343
## 1008 4312.119
## 1009 4320.930
## 1010 4322.402
## 1011 4326.824
## 1012 4346.091
## 1013 4353.547
## 1014 4355.042
## 1015 4359.531
## 1016 4364.030
## 1017 4370.043
## 1018 4380.607
## 1019 4394.267
## 1020 4404.952
## 1021 4417.229
## 1022 4418.769
## 1023 4420.310
## 1024 4423.395
## 1025 4426.485
## 1026 4451.365
## 1027 4454.496
## 1028 4459.202
## 1029 4460.772
## 1030 4468.644
```

```
## 1031 4470.222
## 1032 4473.381
## 1033 4481.300
## 1034 4482.888
## 1035 4492.438
## 1036 4505.240
## 1037 4511.671
## 1038 4514.894
## 1039 4526.214
## 1040 4527.836
## 1041 4535.966
## 1042 4544.129
## 1043 4547.403
## 1044 4552.324
## 1045 4555.611
## 1046 4565.504
## 1047 4590.448
## 1048 4592.122
## 1049 4595.474
## 1050 4603.877
## 1051 4607.248
## 1052 4612.316
## 1053 4622.488
## 1054 4629.298
## 1055 4649.867
## 1056 4672.388
## 1057 4686.373
## 1058 4689.885
## 1059 4695.164
## 1060 4712.860
## 1061 4761.436
## 1062 4765.082
## 1063 4766.907
## 1064 4790.789
## 1065 4794.488
## 1066 4796.341
## 1067 4814.960
## 1068 4820.580
## 1069 4833.754
## 1070 4858.456
## 1071 4873.810
## 1072 4910.760
## 1073 4912.724
## 1074 4920.599
## 1075 4922.573
## 1076 4926.526
## 1077 4938.433
## 1078 4944.413
## 1079 4948.410
## 1080 4950.411
## 1081 4960.449
## 1082 4974.587
## 1083 4976.615
## 1084 4984.748
```

```
## 1085 4999.061
## 1086 5011.411
## 1087 5023.839
## 1088 5038.436
## 1089 5053.141
## 1090 5055.251
## 1091 5059.477
## 1092 5089.311
## 1093 5097.918
## 1094 5115.243
## 1095 5123.962
## 1096 5128.336
## 1097 5132.719
## 1098 5141.515
## 1099 5145.927
## 1100 5148.137
## 1101 5206.459
## 1102 5213.298
## 1103 5220.161
## 1104 5224.749
## 1105 5227.046
## 1106 5247.845
## 1107 5268.860
## 1108 5294.842
## 1109 5297.220
## 1110 5304.372
## 1111 5313.947
## 1112 5321.158
## 1113 5328.395
## 1114 5330.812
## 1115 5345.379
## 1116 5369.889
## 1117 5377.298
## 1118 5409.718
## 1119 5422.323
## 1120 5427.387
## 1121 5429.923
## 1122 5432.463
## 1123 5435.005
## 1124 5442.652
## 1125 5450.326
## 1126 5468.343
## 1127 5473.520
## 1128 5483.911
## 1129 5491.738
## 1130 5520.688
## 1131 5528.652
## 1132 5547.354
## 1133 5550.039
## 1134 5558.114
## 1135 5560.813
## 1136 5568.930
## 1137 5574.358
## 1138 5590.725
```

```
## 1139 5601.707
## 1140 5609.980
## 1141 5612.745
## 1142 5649.012
## 1143 5651.827
## 1144 5680.180
## 1145 5703.130
## 1146 5726.322
## 1147 5740.942
## 1148 5761.573
## 1149 5776.428
## 1150 5836.851
## 1151 5839.915
## 1152 5842.983
## 1153 5908.391
## 1154 5943.411
## 1155 6015.073
## 1156 6065.215
## 1157 6082.167
## 1158 6095.815
## 1159 6109.540
## 1160 6119.886
## 1161 6130.277
## 1162 6140.712
## 1163 6154.695
## 1164 6161.717
## 1165 6172.287
## 1166 6182.904
## 1167 6193.567
## 1168 6215.031
## 1169 6243.943
## 1170 6280.560
## 1171 6313.977
## 1172 6325.214
## 1173 6359.228
## 1174 6401.421
## 1175 6405.291
## 1176 6444.312
## 1177 6448.246
## 1178 6516.042
## 1179 6520.085
## 1180 6524.133
## 1181 6532.249
## 1182 6540.389
## 1183 6552.646
## 1184 6602.236
## 1185 6618.969
## 1186 6648.500
## 1187 6678.351
```

Hypothesis Testing and Confidence Intervals under Heteroskedasticity

1. Carry out a hypothesis test that $\beta_{\text{Budget}} = 0$. Report the $p-$value and draw an appropriate conclusion.

```r
options(scipen = 5)
summary(adj.ad.lm)$tTable
```

```
##                                Value   Std.Error      t-value       p-value
## (Intercept)             3994.3229394 12.64210860 315.95385436  0.000000e+00
## Budget                     1.0051835  0.02641698  38.05066341 3.317197e-196
## AdTypeOutdoor Ads          8.1031987 16.54577579   0.48974426  6.244229e-01
## AdTypeRadio and Podcasts  10.5905337 16.71422721   0.63362389  5.264721e-01
## AdTypeSocial Media Ads     0.3641742 16.22705127   0.02244241  9.820996e-01
## AdTypeVideo Ads           10.5673467 17.13964037   0.61654425  5.376766e-01
```

The p-value for $H_0 : \beta_{\text{Budget}} = 0$ is `3.317197e-196`. We reject the null hypothesis and conclude that β_{Budget} does NOT equal 0, i.e., that `Budget` has a non-zero coefficient in the model, or that `Budget` has a non-zero effect on the number of `Visits`.

2. Carry out a hypothesis test that $\beta_{\text{Budget}} = 1$. Report the $p-$value and draw an appropriate conclusion.

```r
a.matrix <- c(0, 1, 0, 0, 0, 0)
mytest <- glht(adj.ad.lm, linfct = t(a.matrix), rhs=1)
summary(mytest)
```

```
##
##   Simultaneous Tests for General Linear Hypotheses
##
## Fit: gls(model = Visits ~ Budget + AdType, data = data, weights = varExp(form = ~Budget),
##     method = "ML")
##
## Linear Hypotheses:
##        Estimate Std. Error z value Pr(>|z|)
## 1 == 1  1.00518    0.02642   0.196    0.844
## (Adjusted p values reported -- single-step method)
```

The p-value for $H_0 : \beta_{\text{Budget}} = 1$ is `0.844`. We fail to reject the null hypothesis and conclude that there is insufficient evidence that $\beta_{\text{Budget}} = 1$ is not true, i.e., that a 1 unit increase in `Budget` has an effect on `Visits`.

3. Carry out a test that the `AdType` has no effect on the visits.

```r
reduced.gls <- gls(model= Visits~Budget, data= data,
                   weights=varExp(form=~Budget), method="ML")
anova(adj.ad.lm, reduced.gls)
```

```
##             Model df      AIC      BIC    logLik   Test   L.Ratio p-value
## adj.ad.lm       1  8 13740.39 13779.65 -6862.192
## reduced.gls     2  4 13733.17 13752.80 -6862.585 1 vs 2 0.7853927  0.9404
```

The p-value for the test to determine if `AdType` has an effect on `Visits`, or $H_0 : \beta_{\text{AdType}} = 0$, is `0.9404`. We fail to reject the null hypothesis and conclude there is insufficient evidence that $\beta_{\text{AdType}} = 0$ is not true, i.e., that `AdType` has an effect on `Visits`.

4. Construct a 95% confidence interval for β_{Budget}.

```r
confint(adj.ad.lm, level = 0.95)
```

```
##                                2.5 %     97.5 %
## (Intercept)             3969.5448619 4019.10102
## Budget                     0.9534072    1.05696
## AdTypeOutdoor Ads        -24.3259260   40.53232
## AdTypeRadio and Podcasts -22.1687497   43.34982
## AdTypeSocial Media Ads   -31.4402619   32.16861
## AdTypeVideo Ads          -23.0257311   44.16042
```

We are 95% confident that the value of β_{Budget} ranges from 0.9534 to 1.0569.

5. Construct a 95% confidence interval for θ in your variance function. Draw a conclusion about the variability of `Visits` as a function of `Budget` (this answer research question #2).

```r
#coef(adj.ad.lm$modelStruct, unconstrained=FALSE)
intervals(adj.ad.lm, level=0.95)
```

```
## Approximate 95% confidence intervals
##
##  Coefficients:
##                                lower         est.       upper
## (Intercept)              3969.514654 3994.3229394 4019.131225
## Budget                      0.953344    1.0051835    1.057023
## AdTypeOutdoor Ads         -24.365461    8.1031987   40.571859
## AdTypeRadio and Podcasts  -22.208687   10.5905337   43.389755
## AdTypeSocial Media Ads    -31.479036    0.3641742   32.207384
## AdTypeVideo Ads           -23.066685   10.5673467   44.201379
##
##  Variance function:
##             lower        est.       upper
## expon 0.001833602 0.002014862 0.002196122
##
##  Residual standard error:
##    lower     est.    upper
## 68.53122 76.40797 85.19004
```

We are 95% confident that the variability of `Visits` is between between a value of 0.0018 and 0.0022, which is difficult to interpret, but because that range is positive, we are 95% confident that as `Budget` increases, the variability of `Visits` increase.

Calculating Standard Errors

1. Calculate the standard error of $\hat{\beta}_{\text{Budget}}$ in the *iid* linear regression model for `Visits` by using the result shown at the end of the lecture slides. Verify this is the same thing that `lm()` is doing to calculate standard errors.

```r
#adj.ad.lm
formula = Visits~Budget+AdType
X <- model.matrix(object=formula, data=data[c(3,2,4)])
y <- data[["Visits"]]

B <- solve(t(X)%*%X)%*%t(X)%*%y
sigma_hat <- mean(summary(ad.lm)$residuals^2)

var_beta_hat <- sigma_hat*solve(t(X)%*%X)
std.error <- sqrt(diag(var_beta_hat))[2] #yes, it is the same! #0.03210742
std.error
```

```
##     Budget
## 0.03210742
```

```r
summary(ad.lm) #0.0322
```

```
##
## Call:
## lm(formula = Visits ~ . - Company, data = data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1313.54  -143.98     3.61   140.65  1095.48
##
## Coefficients:
##                          Estimate Std. Error t value Pr(>|t|)
## (Intercept)             3995.4368    26.0962 153.104   <2e-16 ***
## Budget                     1.0167     0.0322  31.571   <2e-16 ***
## AdTypeOutdoor Ads         17.6226    28.9569   0.609    0.543
## AdTypeRadio and Podcasts  31.7844    29.0028   1.096    0.273
## AdTypeSocial Media Ads   -40.2877    29.3655  -1.372    0.170
## AdTypeVideo Ads          -10.3677    28.7366  -0.361    0.718
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 293 on 994 degrees of freedom
## Multiple R-squared:  0.506,  Adjusted R-squared:  0.5035
## F-statistic: 203.6 on 5 and 994 DF,  p-value: < 2.2e-16
```

The standard error for $\hat{\beta}_{\text{Budget}}$ in the *iid* linear regression model for is 0.03210742, which I checked against the summary of the model, which produced 0.0322.

2. Using the standard error you calculated above, calculate the $t-$statistic $t = \hat{\beta}_{\text{Budget}}/\text{SE}(\hat{\beta}_{\text{Budget}})$ and the corresponding *two-sided* $p-$value (hint: you will use `pt(test.statistic, df=n-P-1)` to get the $p-$value). Verify that this is the same $p-$value as you would get from the `summary()` output from an `lm()` object.

```r
test.stat <- B[2] / std.error
test.stat
```

```
##   Budget
## 31.66575
```

```r
2 * pt(test.stat, df=994, lower.tail = FALSE)
```

```
##        Budget
## 9.902212e-153
```

```r
#check p-value
#summary(ad.lm)$coefficients[2,4]
summary(ad.lm)$coefficients
```

```
##                              Estimate  Std. Error     t value      Pr(>|t|)
## (Intercept)               3995.436823 26.09620297 153.1041442  0.000000e+00
## Budget                       1.016705  0.03220417  31.5706060 4.443485e-152
## AdTypeOutdoor Ads           17.622634 28.95687021   0.6085822  5.429405e-01
## AdTypeRadio and Podcasts    31.784378 29.00280411   1.0959071  2.733849e-01
## AdTypeSocial Media Ads     -40.287672 29.36553423  -1.3719373  1.703925e-01
## AdTypeVideo Ads            -10.367657 28.73657972  -0.3607826  7.183387e-01
```

Both test statistics and p-values are roughly similar, being both so close to 0 as to not require greater accuracy.

3. Using the standard error your calculated above, calculate a 95% confidence interval for β_{Budget} via the formula $\hat{\beta}_{\text{Budget}} \pm t^{*}\text{SE}(\hat{\beta}_{\text{Budget}})$ (hint: you will need `qt(1-0.05/2, df=n-P-1)` to get the t^{*}value). Verify that this is the same interval as you get from using `confint()`.

```r
tstar.stat <- qt(1-0.05/2, df=994) #1.962353
budget.lwr.ci <- B[2] - tstar.stat * std.error
budget.upr.ci <- B[2] + tstar.stat * std.error
print(c(budget.lwr.ci, budget.upr.ci))
```

```
##    Budget    Budget
## 0.9536992 1.0797114
```

```r
#verify conf int
confint(ad.lm, level = 0.95)
```

```
##                                 2.5 %      97.5 %
## (Intercept)              3944.2268499 4046.646797
## Budget                      0.9535093    1.079901
## AdTypeOutdoor Ads         -39.2009793   74.446248
## AdTypeRadio and Podcasts  -25.1293745   88.698130
## AdTypeSocial Media Ads    -97.9132287   17.337885
## AdTypeVideo Ads           -66.7589827   46.023669
```