LiDAR Estimation of Forest Biomass

The following report addresses four questions from the exercise protocol and discusses the methods used. Section 1 provides a brief summary of the history of LiDAR applications in forestry science. In the following sections (2-4), details are discussed of the data and methods used to develop eight candidate regression models. In section 3, these models are evaluated using accuracy assessments and k-fold cross-validation tests.

1. LiDAR tools in forestry science

Since the 1980s, LiDAR tools have been applied to forest inventories (1–6). These have two approaches to remote sensing: area-based approaches and individual tree detection. Based on remote measurements of mean tree diameter and mean tree height, studies have used LiDAR data to estimate forest biomass (7–10), examine forest structure (11–13), and examine individual tree characteristics such as crown area and length, stem location (14–16), canopy fuel parameters (17), and even tree species (18,19). However, integrating remote sensing data into a multi-phase forest inventory sampling requires a statistical framework that is based on unbiased model-assisted estimators (20–22).

2. Building a LiDAR model for predicting above ground biomass (AGB)

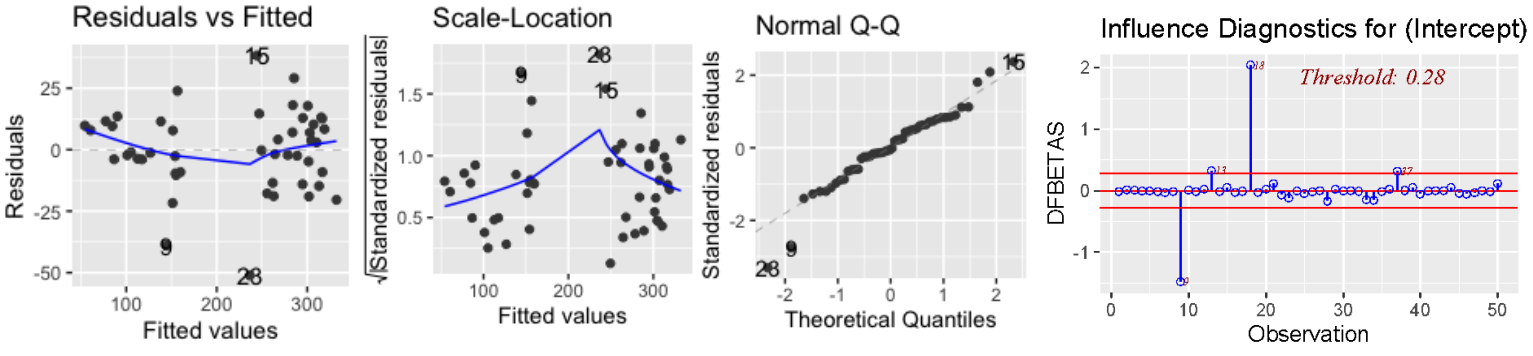
The aim of this exercise was to develop and evaluate regression models for predicting AGB of the Clocaenog forest site. Models were calibrated with data between field plot measurements and remote sensing statistics, which were linked using global navigation satellite systems (23). Field data was collected from among 50 randomly sampled concentric circular plots (10m radius), including measurements of tree diameters at breast height (dbh, cm). Using allometric equations, dbh measurements were used to estimate a response variable: plot-level above ground biomass (*AGB*, Mg/ha^{-1}). Remote sensing data was generated from an airborne laser scanning survey (ALS). From this survey, LiDAR returns (Light Detection and Ranging) were calculated to create three predictor variables. These included: mean tree height (MeanH) based on height returns (11,24–29), standard deviation (SD) of height returns, and fractional canopy height (Cover) based on ratio of height returns above 1.3m to total returns (30–35).

Table 1. Descriptive statistics of predictor variables.

Variable	Mean	SD	SE	Skewness	Kurtosis	SW
MeanH	13.05	5.77	0.82	-0.32	-1.40	***
Cover	0.92	0.08	0.01	-1.76	4.03	***
SD	5.41	2.84	0.40	-0.15	-1.47	***

Table 2. Phase 1 diagnostics; Hypothesis tests of 8 cross-validated regressions of simple linear (LM), stepwise linear models (LM.Stepwise), nonlinear least squared (NLS), stepwise nonlinear least squared regression (NLS.Stepwise), and robust regression (RR) using full dataset and cleaned dataset with influential outliers removed.

Model	SW	BP	DW	BF	Variance Inflation Factor			BIC	AIC
					MeanH	SD	Cover		
1.LM.Full	0.131	0.637	0.345	0.025	3.234	5.451	2.886	437.286	427.727
2.LM.Cleaned	0.949	0.187	0.303	0.001	3.743	4.951	9.493	413.282	403.926
3.LM.Stepwise	0.916	0.586	0.437	0.010	1.903	1.903	/	442.689	435.041
4.NLS.Full	0.968	0.417	0.344	0.049	28.554	12.128	2.959	439.239	427.766
5.NLS.Cleaned	0.961	0.110	0.287	0.002	34.967	13.391	5.221	415.310	404.083
6.RR.Huber	0.002	0.002	0.345	0.025	3.234	5.451	2.886	429.647	430.087
7.RR.Tukey	0.000	0.002	0.345	0.25	3.234	5.451	2.886	444.975	435.415
8.RR.Hampel	0.005	0.002	0.345	0.25	3.234	5.451	2.886	438.499	428.938



1. Accuracy assessments of LiDAR AGB models

Mean Biomass (full/cleaned): 220.628/223.965

Model	RMSE	RMSE%	MAE	MAE%	R^2	Theil's U	$G_4^{2\ change}$
1.LM.Full	16.63	7.6%	13.67	6.2%	0.971	204.25	
2.LM.Cleaned	15.92	7.1%	13.15	5.9 %	0.970	41.64	
3.LM.Stepwise	16.44	7.5%	13.34	6.1%	0.973		
4.NLS.Full	16.34	7.4%	13.26	6.0%	0.974		
5.NLS.Cleaned	15.90	7.1%	13.91	6.2%	0.971		
6.RR.Huber	17.19	7.8%	14.08	6.4%	0.970		
7.RR.Tukey	17.31	7.9%	13.94	6.3%	0.967		
8.RR.Hampel	17.48	7.9%	14.08	6.4%	0.972		

Model estimates for final equation	MeanH	SD	Cover
1.LM.Full	9.75(0.73)***	15.56(1.93)***	157.69(51.38)**
2.LM.Cleaned	9.73(0.75)***	15.10(2.39)***	132.09(89.79)
3.LM.Stepwise	11.19(0.61)***	10.77(1.24)***	/
4.NLS.Full	7.03(2.16)**	12.72(2.86)***	146.78(51.59)
5.NLS.Cleaned	6.95(2.27)**	13.23(2.82)***	159.11(91.51)
6.RR.Huber	9.53(0.60)***	15.49(1.58)***	116.03(42.07)**
7.RR.Tukey	9.39(0.58)***	15.42(1.53)***	80.29(40.51)
8.RR.Hampel	9.59(0.66)***	15.60(1.73)***	129.46(46.00)**

Table 2: #Durban-Watson test for autocorrelation of disturbances. Shaprio-Wilks's p-values reporting signficant deviation from normal distribution at the levels of <0.05, 0.01, 0.001 as *, ** and **

From Tojal et al 2019: Shapiro–Wilk test to verify the normality of the residuals, Breusch–Pagan test to analyze the homoscedasticity of the residuals, Durbin–Watson test to detect possible dependencies between the data, Variance Inflation Factor (VIF) to detect collinearity problems in the model [37], Ramsey´s RESET linearity test to verify linear relations, and, finally, Bonferroni´s test to find statistically significant atypical values. All tests, except VIF, were calculated at the 95% level of significance. (GN: Gauss-Newton method is an iterative algorithm to solve non-linear least squares problems –inspired by kernel regression and Newton method but does not require computation of approximate Hessian, instead looks for a minimum sum of squared function values (36).

2. Limits to LiDAR biomass models

Based on observations in the literature that upper percentile heights typically correlate with plot-level tree height (e.g. Magnussen & Boudewyn, 1998; Næsset, 1997), all the major upper height percentiles from the median to max should be extracted and correlated to field height values. Using pearson's rank correlations, links to be made between plot-level field-measured mean tree height and all percentile heights should be performed and dated (Hopkinson)

Influence of Vegetation Structure on Lidar-derived Canopy Height and Fractional Cover in Forested Riparian Buffers During Leaf-Off and Leaf-On Conditions (Wasser et al)

The R^2 is a biased measure, inasmuch it increases the more terms are added into the model as predictors. The adjusted R^2 compensates for this overfitting by penalizing for the number of terms included as predictors

Bibliography

1. Garcia M, Saatchi S, Ustin S, Balzter H. Modelling forest canopy height by integrating airborne LiDAR samples with satellite Radar and multispectral imagery. Int J Appl earth Obs Geoinf. 2018;66:159–73.

2. Guo Z, Chi H, Sun G. Estimating forest aboveground biomass using HJ-1 Satellite CCD and ICESat GLAS waveform data. Sci China Earth Sci. 2010;53(1):16–25.

3. Næsset E, Gobakken T, Solberg S, Gregoire TG, Nelson R, Ståhl G, et al. Model-assisted regional forest biomass estimation using LiDAR and InSAR as auxiliary data: A case study from a boreal forest area. Remote Sens Environ. 2011;115(12):3599–614.

4. Nelson R, Margolis H, Montesano P, Sun G, Cook B, Corp L, et al. Lidar-based estimates of aboveground biomass in the continental US and Mexico using ground, airborne, and satellite observations. Remote Sens Environ. 2017;188:127–40.

5. Nelson R, Krabill W, MacLean G. Determining forest canopy characteristics using airborne laser data. Remote Sens Environ. 1984;15(3):201–12.

6. Maclean GA, Krabill WB. Gross-merchantable timber volume estimation using an airborne LIDAR system. Can J Remote Sens. 1986;12(1):7–18.

7. Ene LT, Gobakken T, Andersen H-E, Næsset E, Cook BD, Morton DC, et al. Large-area hybrid estimation of aboveground biomass in interior Alaska using airborne laser scanning data. Remote Sens Environ. 2018;204:741–55.

8. Laurin GV, Puletti N, Chen Q, Corona P, Papale D, Valentini R. Above ground biomass and tree species richness estimation with airborne lidar in tropical Ghana forests. Int J Appl earth Obs Geoinf. 2016;52:371–9.

9. Magnussen S, Nord-Larsen T, Riis-Nielsen T. Lidar supported estimators of wood volume and aboveground biomass from the Danish national forest inventory (2012--2016). Remote Sens Environ. 2018;211:146–53.

10. Shao G, Shao G, Gallion J, Saunders MR, Frankenberger JR, Fei S. Improving Lidar-based aboveground biomass estimation of temperate hardwood forests with varying site productivity. Remote Sens Environ. 2018;204:872–82.

11. Næsset E. Predicting forest stand characteristics with airborne scanning laser using a practical two-stage procedure and field data. Remote Sens Environ. 2002;80(1):88–99.

12. Nelson R, Oderwald R, Gregoire TG. Separating the ground and airborne laser sampling phases to estimate tropical forest basal area, volume, and biomass. Remote Sens Environ. 1997;60(3):311–26.

13. Nilsson M. Estimation of tree heights and stand volume using an airborne lidar system. Remote Sens Environ. 1996;56(1):1–7.

14. Gougeon FA, Leckie DG, others. Forest information extraction from high spatial resolution images using an individual tree crown approach. Pacific Forestry Centre; 2002.

15. Reutebuch SE, Andersen H-E, McGaughey RJ. Light detection and ranging (LIDAR): an emerging tool for multiple resource inventory. J For. 2005;103(6):286–92.

16. Lee H, Slatton KC, Roth BE, Cropper Jr WP. Adaptive clustering of airborne LiDAR data to segment individual tree crowns in managed pine forests. Int J Remote Sens. 2010;31(1):117–39.

17. Andersen H-E, McGaughey RJ, Reutebuch SE. Estimating forest canopy fuel parameters using LIDAR data. Remote Sens Environ. 2005;94(4):441–9.

18. Shi Y, Wang T, Skidmore AK, Heurich M. Important LiDAR metrics for discriminating forest tree species in Central Europe. ISPRS J Photogramm Remote Sens. 2018;137:163–74.

19. Brandtberg T. Classifying individual tree species under leaf-off and leaf-on conditions using airborne lidar. ISPRS J Photogramm Remote Sens. 2007;61(5):325–40.

20. Tojal L-T, Bastarrika A, Barrett B, Sanchez Espeso JM, Lopez-Guede JM, Graña M. Prediction of Aboveground Biomass from Low-Density LiDAR Data: Validation over P. radiata Data from a Region North of Spain. Forests. 2019;10(9):819.

21. Corona P. Integration of forest mapping and inventory to support forest management. iForest-Biogeosciences For. 2010;3(3):59.

22. Corona P. Consolidating new paradigms in large-scale monitoring and assessment of forest ecosystems. Environ Res. 2016;144:8–14.

23. Valbuena R, Mauro F, Rodr\'\iguez-Solano R, Manzanera JA. Partial least squares for discriminating variance components in global navigation satellite systems accuracy obtained under scots pine canopies. For Sci. 2012;58(2):139–53.

24. Hopkinson C, Chasmer L, Hall RJ. The uncertainty in conifer plantation growth prediction from multi-temporal lidar datasets. Remote Sens Environ. 2008;112(3):1168–80.

25. Magnussen S, Boudewyn P. Derivations of stand heights from airborne laser scanner data with canopy-based quantile estimators. Can J For Res. 1998;28(7):1016–31.

26. Means JE, Acker SA, Fitt BJ, Renslow M, Emerson L, Hendrix CJ, et al. Predicting forest stand characteristics with airborne scanning lidar. Photogramm Eng Remote Sensing. 2000;66(11):1367–72.

27. Naesset E. Determination of mean tree height of forest stands using airborne laser scanner data. ISPRS J Photogramm Remote Sens. 1997;52(2):49–56.

28. Næsset E, Økland T. Estimating tree height and tree crown properties using airborne scanning laser in a boreal nature reserve. Remote Sens Environ. 2002;79(1):105–15.

29. Popescu SC. Estimating biomass of individual pine trees using airborne lidar. Biomass and Bioenergy. 2007;31(9):646–55.

30. Hopkinson C, Chasmer L, Lim K, Treitz P, Creed I. Towards a universal lidar canopy height indicator. Can J Remote Sens. 2006;32(2):139–52.

31. Hopkinson C, Chasmer L. Testing LiDAR models of fractional cover across multiple forest ecozones. Remote Sens Environ. 2009;113(1):275–88.

32. Morsdorf F, Koetz B, Meier E, Itten KI, Allgöwer B. The potential of discrete return, small footprint airborne laser scanning data for vegetation density estimation. Proc ISPRS WG III/3. 2005;3(4):3.

33. Hill RA, Broughton RK. Mapping the understorey of deciduous woodland from leaf-on and leaf-off airborne LiDAR data: A case study in lowland Britain. ISPRS J Photogramm Remote Sens. 2009;64(2):223–33.

34. Wasser L, Day R, Chasmer L, Taylor A. Influence of vegetation structure on lidar-derived canopy height and fractional cover in forested riparian buffers during leaf-off and leaf-on conditions. PLoS One. 2013;8(1).

35. Morsdorf F, Kötz B, Meier E, Itten KI, Allgöwer B. Estimation of LAI and fractional cover from small footprint airborne laser scanning data based on gap fraction. Remote Sens Environ. 2006;104(1):50–61.

36. Cai T, Gao R, Hou J, Chen S, Wang D, He D, et al. A gram-gauss-newton method learning overparameterized deep neural networks for regression problems. arXiv Prepr arXiv190511675. 2019;

LiDAR Estimation of Forest Biomass

R Syntax:

```
#### MSc Inventory Monitoring and Assessment ####
######  Assignment 3; LIDAR AGB Predict #########
############ S.Murphy 19-04-2020 ################

#5 steps that follow:
#1.Explore predictors and run linear model using full dataset. Run diagnostics & check violations in residuals
#2.Clean dataset removing biggest outliers
#3.Run 7 alternative models using full & cleaned dataset. Run diagnostics & accuracy assessment
#4.Choose model based on pre/post-diagnostics and accuracy assessment
#5.Plot residuals using benchmark diagnostics


library(readxl)
ClocaenogField <- read_excel("ClocaenogField.xlsx")
View(ClocaenogField)

#Explore data and normality of predictor variables:
describe(ClocaenogField$MEANH)
describe(ClocaenogField$COVER)
describe(ClocaenogField$SD)
shapiro.test(ClocaenogField$MEANH)
shapiro.test(ClocaenogField$COVER)
shapiro.test(ClocaenogField$SD)

#MODEL 1 (LM1): Linear regression
model_lm1 <- lm(BIOMASS ~ MEANH + COVER + SD, data = ClocaenogField)
summary (model_lm1)

#Run model diagnostics and check linear assumptions
ols_test_normality(model_lm1)
ols_test_breusch_pagan(model_lm1)
dwtest(model_lm1)
ols_coll_diag(model_lm1)
outlierTest(model_lm1, data=Duncan)
ols_test_outlier(model_lm1)
ols_vif_tol(model_lm1)
autoplot(model_lm1)
ols_plot_diagnostics(model_lm1)
ols_plot_cooksd_bar(model_lm1)
ols_plot_dfbetas(model_lm1)
ols_plot_resid_lev(model_lm1)
ols_plot_comp_plus_resid(model_lm1, print_plot = TRUE)

#Bonferroni results significant and influential outliers found.
#Create new dataframe removing outliers and re-run linear model.
#Check again for violations and compare model 1 & 2 with non-linear models.

#remove outliers from rows 9 and 18
ClocaenogField_cleaned <- ClocaenogField[-c(9,18),]
```

```
#MODEL 2 (LM_Cleaned): Linear regression using cleaned dataset
model_lm2 <- lm(BIOMASS ~ MEANH + COVER + SD, data = ClocaenogField_cleaned)
summary (model_lm2)

#run model diagnostics again
ols_test_normality(model_lm2)
ols_test_breusch_pagan(model_lm2)
dwtest(model_lm2)
ols_coll_diag(model_lm2)
outlierTest(model_lm2, data=Duncan)
ols_test_outlier(model_lm2)
ols_vif_tol(model_lm2)
autoplot(model_lm2)
ols_plot_diagnostics(model_lm2)
ols_plot_cooksd_bar(model_lm2)
ols_plot_dfbetas(model_lm2)
ols_plot_resid_lev(model_lm2)
ols_plot_comp_plus_resid(model_lm2, print_plot = TRUE)
#same violations found from bonferroni results and variance inflation factors
#resort to stepwise, non-linear and robust regression and compare...

#MODEL 3 (LM-STEP-Full): Stepwise Linear regression using full dataset
# Step-wise backwards 10-fold cross-validation test of model accuracy based on RMSE
# Set seed for reproducibility
# Set up repeated k-fold cross-validation
train.control_lm1 <- trainControl(method = "cv", number = 10)
# Train the model
step.model_lm1 <- train(BIOMASS ~ MEANH + COVER + SD, data = ClocaenogField, method =
"leapBackward", tuneGrid = data.frame(nvmax = 1:3), trControl = train.control_lm1)
step.model_lm1$results
summary(step.model_lm1$finalModel)
# Results suggest best 2-variable model contain MEANH and SD
# Step-wise model using most best 2-variables: MEANH and SD
model_lm_step_full <- lm(BIOMASS ~ MEANH + SD, data = ClocaenogField)
summary(model_lm_step_full)

#run diagnostics
ols_test_normality(model_lm_step_full)
ols_test_breusch_pagan(model_lm_step_full)
dwtest(model_lm_step_full)
ols_coll_diag(model_lm_step_full)
outlierTest(model_lm_step_full, data=Duncan)
ols_test_outlier(model_lm_step_full)
ols_vif_tol(model_lm_step_full)
autoplot(model_lm_step_full)
ols_plot_diagnostics(model_lm_step_full)
ols_plot_cooksd_bar(model_lm_step_full)
ols_plot_dfbetas(model_lm_step_full)
ols_plot_resid_lev(model_lm_step_full)
ols_plot_comp_plus_resid(model_lm_step_full, print_plot = TRUE)
```

```
# MODEL 4 (NLS) Non-linear least squares regression model using full dataset
# Non-linear least squares regression
model_nls1 <- lm(ClocaenogField$BIOMASS ~ ClocaenogField$MEANH + ClocaenogField$COVER + ClocaenogField$SD + I((ClocaenogField$MEANH + ClocaenogField$COVER + ClocaenogField$SD)^2))
summary(model_nls1)

# MODEL 5 (NLS) Non-linear least squares regression model using cleaned dataset
# Non-linear least squares regression
model_nls1_cleaned <- lm(ClocaenogField_cleaned$BIOMASS ~ ClocaenogField_cleaned$MEANH + ClocaenogField_cleaned$COVER + ClocaenogField_cleaned$SD + I((ClocaenogField_cleaned$MEANH + ClocaenogField_cleaned$COVER + ClocaenogField_cleaned$SD)^2))
summary(model_nls1_cleaned)

# MODEL 6 (RLM.Huber) Robust regression using the Huber M-estimator to reduce outlier influence. Though this does not reduce outliers in predictor variables, so Tukey needed below
rr.huber <- rlm(BIOMASS ~ MEANH + COVER + SD, data = ClocaenogField)
summary(rr.huber)
f.robftest(rr.huber,var = "MEANH")
f.robftest(rr.huber, var = "SD")
f.robftest(rr.huber, var = "COVER")


# MODEL 7 (RLM.Tukey) Robust regression using the Tukey M-estimator that assigns a weight of zero to influential outliers
rr.bisquare <- rlm(BIOMASS ~ MEANH + COVER + SD, data = ClocaenogField, psi = psi.bisquare)
summary(rr.bisquare)
f.robftest(rr.bisquare,var = "MEANH")
f.robftest(rr.bisquare, var = "SD")
f.robftest(rr.bisquare, var = "COVER")

# MODEL 8 (RLM.Tukey) Robust regression using the Tukey M-estimator that assigns a weight of zero to influential outliers
rr.hampel <- rlm(BIOMASS ~ MEANH + COVER + SD, data = ClocaenogField, psi = psi.hampel)
summary(rr.hampel)
f.robftest(rr.hampel,var = "MEANH")
f.robftest(rr.hampel, var = "SD")
f.robftest(rr.hampel, var = "COVER")


# Explore altnerative accuracy assessments:
# sidak p value adjustment
ols_test_breusch_pagan(model_nls1, rhs = TRUE, multiple = TRUE, p.adj = 'sidak')
# holm's p value adjustment
ols_test_breusch_pagan(model_nls1, rhs = TRUE, multiple = TRUE, p.adj = 'holm')
# Global test of model assumptions
library(gvlma)
gvmodel <- gvlma(model_lm1)
summary(gvmodel)
gvmodel_del <- deletion.gvlma(gvmodel)
summary(gvmodel_del)
```

```
#compare models
ols_mallows_cp(model_lm1, model_lm2)
ols_fpe(model_lm1)
ols_hsp(model_lm1)

ols_mallows_cp(model_lm1, model_lm2)
ols_mallows_cp(model_lm1, model_lm_step_full)
ols_mallows_cp(model_lm1, model_nls1)
ols_mallows_cp(model_lm1, model_nls1_cleaned)
ols_mallows_cp(model_lm1, rr.huber)
ols_mallows_cp(model_lm1, rr.bisquare)
ols_mallows_cp(model_lm1, rr.hampel)

AIC(model_lm1)
AIC(model_lm2)
AIC(model_lm_step_full)
AIC(model_nls1)
AIC(model_nls1_cleaned)
AIC(rr.huber)
AIC(rr.bisquare)
AIC(rr.hampel)

BIC(model_lm1)
BIC(model_lm2)
BIC(model_lm_step_full)
BIC(model_nls1)
BIC(model_nls1_cleaned)
BIC(rr.huber)
BIC(rr.bisquare)
BIC(rr.hampel)

glance(model_lm1)
glance(model_lm2)
glance(model_lm_step_full)
glance(model_nls1)
glance(model_nls1_cleaned)
glance(rr.huber)
glance(rr.bisquare)
glance(rr.hampel)

#Explore residual diagnostics
ols_plot_resid_box
ols_plot_resid_fit
ols_plot_resid_hist
ols_plot_resid_qq


#big outliers so choose to run a robust regression using MASS pkg:
#from https://stats.idre.ucla.edu/r/dae/robust-regression/
#Robust regression is iterated re-weighted least squares (IRLS). The
#command is rlm in the MASS package. There are several
```

```
#weighting functions that can be used for IRLS. We
#first use the Huber weights and then bi-square weighting. We
#will then look at the final weights created by the IRLS process.

#robust methods due to variance of resids and measures of influence:
#from https://stats.idre.ucla.edu/r/dae/robust-regression/
#We can see that the weight given to Mississippi
#is dramatically lower using the bisquare weighting
#function than the Huber weighting function and
#the parameter estimates from these two different
#weighting methods differ. When comparing the
#results of a regular OLS regression and a robust
#regression, if the results are very different,
#you will most likely want to use the results
#from the robust regression. Large differences
#suggest that the model parameters are being
#highly influenced by outliers. Different functions
#have advantages and drawbacks. Huber weights can
#have difficulties with severe outliers, and
#bisquare weights can have difficulties converging
#or may yield multiple solutions.

#(robust) sandwich variance estimator for linear regression:
#from:
https://thestatsgeek.com/2014/02/14/the-robust-sandwich-variance-estimator-for-linear-regression-using-r/
#This method allowed us to estimate valid standard errors
#for our coefficients in linear regression, without requiring
#the usual assumption that the residual errors have constant variance.


# 10 k-fold cross validation
# define training control
train_control_kfold <- trainControl(method="cv", number=10)
# fix the parameters of the algorithm
grid <- expand.grid(.fL=c(0), .usekernel=c(FALSE))
# train the model
model_lm1_kfold <- train(BIOMASS ~ MEANH + SD + COVER, data=ClocaenogField,
trControl=train_control_kfold)
# summarize results
print(model_lm1_kfold)
View(model_lm1_kfold)
summary(model_lm1_kfold)


# repeated 10 k-fold cross validation
# define training control
train_control_kfold_repeat <- trainControl(method="repeatedcv", number=10, repeats=3)
# train the model
model_lm1_kfold_repeat <- train(BIOMASS ~ MEANH + SD + COVER, data = ClocaenogField, trControl =
train_control_kfold_repeat)
```

```
model_lm2_kfold_repeat <- train(BIOMASS ~ MEANH + SD + COVER, data = ClocaenogField_cleaned, trControl=train_control_kfold_repeat)
model_lm_step_full_kfold_repeat <- train(BIOMASS ~ MEANH + SD, data = ClocaenogField, trControl=train_control_kfold_repeat)
model_nls1_kfold_repeat <- train(BIOMASS ~ MEANH + SD + COVER, data = ClocaenogField, trControl=train_control_kfold_repeat)
model_nls1_cleaned_kfold_repeat <- train(BIOMASS ~ MEANH + SD + COVER, data = ClocaenogField_cleaned, trControl=train_control_kfold_repeat)
rr.huber_kfold_repeat <- train(BIOMASS ~ MEANH + SD + COVER, data = ClocaenogField, trControl=train_control_kfold_repeat)
rr.bisquare_kfold_repeat <- train(BIOMASS ~ MEANH + SD + COVER, data = ClocaenogField, trControl=train_control_kfold_repeat)
rr.hampel_kfold_repeat <-train(BIOMASS ~ MEANH + SD + COVER, data = ClocaenogField, trControl=train_control_kfold_repeat)

print(model_lm1_kfold_repeat)
print(model_lm2_kfold_repeat)
print(model_lm_step_full_kfold_repeat)
print(model_nls1_kfold_repeat)
print(model_nls1_cleaned_kfold_repeat)
print(rr.huber_kfold_repeat)
print(rr.bisquare_kfold_repeat)
print(rr.hampel_kfold_repeat)

# Leave-one-out cross validation
# define training control
train_control_LOOVC <- trainControl(method="LOOCV")
# train the model
model_lm1_LOOVC <- train(BIOMASS ~ MEANH + SD + COVER, data = ClocaenogField, trControl=train_control_LOOVC)
model_lm2_LOOVC <- train(BIOMASS ~ MEANH + SD + COVER, data = ClocaenogField_cleaned, trControl=train_control_LOOVC)
model_lm_step_full_LOOVC <- train(BIOMASS ~ MEANH + SD, data = ClocaenogField, trControl=train_control_LOOVC)
model_nls1_LOOVC <- train(BIOMASS ~ MEANH + SD + COVER, data = ClocaenogField, trControl=train_control_LOOVC)
model_nls1_cleaned_LOOVC <- train(BIOMASS ~ MEANH + SD + COVER, data = ClocaenogField_cleaned, trControl=train_control_LOOVC)
rr.huber_LOOVC <- train(BIOMASS ~ MEANH + SD + COVER, data = ClocaenogField, trControl=train_control_LOOVC)
rr.bisquare_LOOVC <- train(BIOMASS ~ MEANH + SD + COVER, data = ClocaenogField, trControl=train_control_LOOVC)
rr.hampel_LOOVC <-train(BIOMASS ~ MEANH + SD + COVER, data = ClocaenogField, trControl=train_control_LOOVC)

print(model_lm1_LOOVC)
summary(model_lm1_LOOVC)
View(model_lm1_LOOVC)

print(model_lm2_LOOVC)
print(model_lm_step_full_LOOVC)
```

```
print(model_nls1_LOOVC)
print(model_nls1_cleaned_LOOVC)
print(rr.huber_LOOVC)
print(rr.bisquare_LOOVC)
print(rr.hampel_LOOVC)

#run diagnostics
shapiro.test(resid(model_lm1_kfold_repeat))
shapiro.test(resid(model_lm2_kfold_repeat))
shapiro.test(resid(model_lm_step_full_kfold_repeat))
shapiro.test(resid(model_nls1_kfold_repeat))
shapiro.test(resid(model_nls1_cleaned_kfold_repeat))
shapiro.test(resid(rr.huber_kfold_repeat))
shapiro.test(resid(rr.bisquare_kfold_repeat))
shapiro.test(resid(rr.hampel_kfold_repeat))

bptest(model_lm1_kfold_repeat)
bptest(model_lm2_kfold_repeat)
bptest(model_lm_step_full_kfold_repeat)
bptest(model_nls1_kfold_repeat)
bptest(model_nls1_cleaned_kfold_repeat)
bptest(rr.huber_kfold_repeat)
bptest(rr.bisquare_kfold_repeat)
bptest(rr.hampel_kfold_repeat)

dwtest(rr.huber)
dwtest(rr.bisquare)
dwtest(rr.hampel)

outlierTest(lm(rr.huber))
outlierTest(lm(rr.bisquare))
outlierTest(lm(rr.hampel))

vif(rr.huber)
vif(rr.bisquare)
vif(rr.hampel)

AIC(model_lm1)
AIC(model_lm2)
AIC(model_lm_step_full)
AIC(model_nls1)
AIC(model_nls1_cleaned)
AIC(rr.huber)
AIC(rr.bisquare)
AIC(rr.hampel)

BIC(model_lm1)
BIC(model_lm2)
BIC(model_lm_step_full)
BIC(model_nls1)
BIC(model_nls1_cleaned)
```

```
BIC(rr.huber)
BIC(rr.bisquare)
BIC(rr.hampel)

TheilU(ClocaenogField$BIOMASS, model_lm1, type=1)
theil.wtd(model_lm1, weights = NULL)


bias_training <- rnorm(40, 2, sd = 0.5)
bias(bias_training, ClocaenogField)
bias(samp, pop)
bias(samp, pop, type = 'relative')
bias(samp, pop, type = 'standardized')
dev <- samp - pop
bias(dev)
nom.uncertainty(model_lm1)
lambda3(model_nls1)
CronbachAlpha(model_lm1)
guttman(model_lm1, missing = "complete", standardize = FALSE)


ols_plot_response(model_lm1)

u_theil_train <- createDataPartition(ClocaenogField$BIOMASS, p=0.5, list = FALSE)
u_theil_trainingData <- ClocaenogField[u_theil_train]
u_theil_testData <- historical.df[-train,]

dim(trainingData)
```