

* Encoding: UTF-8.

variable labels parcel_id 'Parcel Number'.
variable labels zone 'Zoning'.
variable labels nbhd 'Neighborhood'.
variable labels parcel_acres 'Total Parcel Size'.
variable labels non_valued_acres 'Unuseable Parcel Acres'.
variable labels crop_value 'Crop Value'.
variable labels detached_value 'Detached Value'.
variable labels flood_plain 'Floodplain'.
variable labels topography 'Topography'.
variable labels street_finsh 'Street Finish'.
variable labels gas 'Natural Gas Hookup'.
variable labels curbs 'Curbing'.
variable labels side_walk 'Sidewalk'.
variable labels water_shed 'County Water Shed'.
variable labels fire_risk 'Fire Risk'.
variable labels assessed_land 'Previous Land Value'.
variable labels assessed_imp 'Previous Imp. Value'.
variable labels assessed_total 'Previous Assessed Total'.
variable labels sale_date 'Sale Date'.
variable labels sale_price 'Sale Price'.
variable labels parcel_lots 'Lots Sold'.
variable labels lot_acreage 'Acreage of Lots Sold'.

variable labels A1H 'Apple 1 High Density'.
variable labels A1S 'Apple 1 Standard Density'.
variable labels A2H 'Apple 2 High Density'.
variable labels A2S 'Apple 2 Standard Density'.
variable labels A3H 'Apple 3 High Density'.
variable labels A3S 'Apple 3 Standard Density'.
variable labels CHH 'Cherries High Density'.
variable labels CHS 'Cherries Standard Density'.
variable labels HOP 'Hops'.
variable labels IRR 'Irrigation'.
variable labels JGR 'Juice Grapes'.
variable labels PEH 'Pears High Density'.
variable labels PES 'Pears Standard Density'.
variable labels SFH 'Stone Fruit High Density'.
variable labels SFS 'Stone Fruit Standard Density'.
variable labels WGR 'Wine Grapes'.

compute assessed_imp = crop_value + detached_value.
compute assessed_total = assessed_land + assessed_imp.

compute useable_acres = parcel_acres - non_valued_acres.
variable labels useable_acres 'Useable Parcel Size'.

compute non_lot_acres = parcel_acres - non_valued_acres - lot_acreage.
if (non_lot_acres <= .01) non_lot_acres = 0.
variable labels non_lot_acres 'Land not in Lots'.

COMPUTE SAR=assessed_total/sale_price.
variable labels SAR 'Sales Assessment Ratio'.

```
compute Ratio_Outlier = 1.
if (SAR < .25) Ratio_Outlier = 0.
if (SAR > 1.75) Ratio_Outlier = 0.
filter by Ratio_Outlier.
```

```
compute smonth = xdate.month(sale_date).
compute syear = 2025 - xdate.year(sale_date).
```

```
COMPUTE MONTHS=SMONTH.
variable labels Months 'Months Prior to Appraisal Date'.
if (SYEAR=2) MONTHS=SMONTH+12.
If (SYEAR=3) MONTHS=SMONTH+24.
If (SYEAR=4) MONTHS=SMONTH+36.
If (SYEAR=5) MONTHS=SMONTH+48.
If (SYEAR=6) MONTHS=SMONTH+60.
FORMATS MONTHS(F4.0).
MEANS MONTHS BY SYEAR BY SMONTH.
```

```
GRAPH/SCATTER=months WITH SAR/Title="Sales Assessment Ratio Change Over Time".
GRAPH/HISTOGRAM=SAR.
```

```
compute acre_range = 0.
if(parcel_acres>= .5) acre_range = 1.
if(parcel_acres>=1) acre_range = 2.
if(parcel_acres>=2) acre_range = 3.
if(parcel_acres>=5) acre_range = 4.
if(parcel_acres>=10) acre_range = 5.
if(parcel_acres>=20) acre_range = 6.
```

```
variable labels acre_range 'Acre Range'.
add value labels acre_range 0 'Up to 0.5' 1 '0.5 to 1' 2 '1 to 2' 3 '2 to 5' 4 '5 to 10' 5 '10 to 20' 6 '20 and up'.
```

```
* Ratio Statistics prior to time adjustment.
RATIO STATISTICS assessed_total WITH sale_price/PRINT=COD MEAN MEDIAN WGTMEAN N.
RATIO STATISTICS assessed_total WITH sale_price BY acre_range (ASCENDING)/PRINT=COD MEAN MEDIAN
WGTMEAN N.
RATIO STATISTICS assessed_total WITH sale_price BY zone (ASCENDING)/PRINT=COD MEAN MEDIAN
WGTMEAN N.
RATIO STATISTICS assessed_total WITH sale_price BY nbhd (ASCENDING)/PRINT=COD MEAN MEDIAN
WGTMEAN N.
RATIO STATISTICS assessed_total WITH sale_price BY property_use_category (ASCENDING)/PRINT=COD
MEAN MEDIAN WGTMEAN N.
```

```
COMPUTE LN_SAR=LN(SAR).
variable labels LN_SAR 'Natural Log of SAR'.
GRAPH/SCATTER=MONTHS WITH LN_SAR.
```

```
REGRESSION
/STATISTICS COEFF OUTS R ANOVA
/CRITERIA=PIN(.05) POUT(.10) TOLERANCE(.0001)
/DEPENDENT LN_SAR
/METHOD=STEPWISE MONTHS.
```

```

COMPUTE END1=24.
COMPUTE END2=36.
COMPUTE MONTHS1=MONTHS.
IF (MONTHS>END1) MONTHS1=END1.
COMPUTE MONTHS2=0.
IF (MONTHS>END1) MONTHS2=MONTHS-END1.
IF (MONTHS>END2) MONTHS2=END2-END1.
COMPUTE MONTHS3=0.
IF (MONTHS>END2) MONTHS3=MONTHS-END2.
MEANS MONTHS1 MONTHS2 MONTHS3 BY MONTHS/CELLS=MEAN.

```

```

REGRESSION
/STATISTICS COEFF OUTS R ANOVA
/CRITERIA=PIN(.05) POUT(.10) TOLERANCE(.0001)
/DEPENDENT LN_SAR
/METHOD=FORWARD MONTHS1 MONTHS2 MONTHS3.

```

* Regression for time by months is not quite as good as the 3 splines. Using splines regression for time adjustment.

```

COMPUTE r1 = 0.00.
COMPUTE r2 = 0.00.
COMPUTE r3 = 0.01356532.
formats r1(f10.8).
formats r2(f10.8).
formats r3(f10.8).

```

```

COMPUTE PRICE_INDEX=(1+r1)**MONTHS1 * (1+r2)**MONTHS2*(1+r3)**MONTHS3.
GRAPH/SCATTER=MONTHS WITH PRICE_INDEX.

```

```

MEANS PRICE_INDEX BY MONTHS.
COMPUTE END_INDEX=1.000.
COMPUTE TAF=PRICE_INDEX/END_INDEX.
GRAPH/SCATTER=MONTHS WITH TAF.
MEANS TAF BY MONTHS.

```

```

*TIME-ADJUSTED PRICES.
COMPUTE TASP=sale_price*TAF.
variable labels TASP 'Time Adjusted Sales Price'.
DES sale_price TASP TAF.
COMPUTE TASAR=assessed_total/TASP.
variable labels TASAR 'Time Adjusted Sales Assessment Ratio'.

```

```

GRAPH/SCATTER=MONTHS WITH TASAR.

```

```

GRAPH/HISTOGRAM=sale_price.
GRAPH/HISTOGRAM=TASP.
GRAPH/HISTOGRAM=SAR.
GRAPH/HISTOGRAM=TASAR.

```

```

compute adj_sale_price = tasp - detached_value.
variable labels adj_sale_price 'Time Adj. Sale Price minus outbuildings'.

```

```
compute sppa = adj_sale_price/useable_acres.  
variable labels sppa 'Sale Price per Acre of Useable Acres'.  
GRAPH/SCATTER=useable_acres WITH adj_sale_price.  
GRAPH/SCATTER=useable_acres WITH sppa.
```

```
compute LNAcres = ln(useable_acres).  
compute LNsppa = ln(sppa).  
graph/scatter=LNAcres with LNsppa.
```

```
compute LNSP = ln(adj_sale_price).
```

```
compute first10LN = 0.  
compute first10 = 0.  
compute excessLand = 0.  
if (useable_acres <10) first10 = useable_acres.  
if (useable_acres >=10) first10 = 10.  
if (useable_acres <10) first10LN = ln(useable_acres).  
if (useable_acres >=10) first10LN = ln(10).  
if (useable_acres >10) excessLand = useable_acres - 10.
```

```
compute excess_land_LN = 0.  
if (excessLand > 1) excess_land_LN = ln(excessLand).
```

* Ratio Statistics after time adjustment.

```
RATIO STATISTICS assessed_total WITH tasp/PRINT=COD MEAN MEDIAN WGTMEAN N.
```

```
RATIO STATISTICS assessed_total WITH tasp BY acre_range (ASCENDING)/PRINT=COD MEAN MEDIAN  
WGTMEAN N.
```

```
RATIO STATISTICS assessed_total WITH tasp BY zone (ASCENDING)/PRINT=COD MEAN MEDIAN WGTMEAN N
```

```
RATIO STATISTICS assessed_total WITH tasp BY nbhd (ASCENDING)/PRINT=COD MEAN MEDIAN WGTMEAN N
```

```
RATIO STATISTICS assessed_total WITH tasp BY property_use_category (ASCENDING)/PRINT=COD MEAN  
MEDIAN WGTMEAN N.
```

```
means TABLES=sppa BY property_use_category /CELLS=MEAN MEDIAN COUNT STDDEV.
```

```
compute AG_USE = 0.
```

```
if (property_use_category = 'Agricultural') AG_USE = 1.
```

```
compute OT_USE = 0.
```

```
if (property_use_category = 'Land/Other') OT_USE = 1.
```

```
compute RES_USE = 0.
```

```
if (property_use_category = 'Residential') RES_USE = 1.
```

```
MEANS TABLES=sppa BY zone /CELLS=MEAN MEDIAN COUNT STDDEV.
```

*Probably exclude AG and R10/%.

```
compute AGZone = 0.
```

```
if (zone = 'AG') AGZone = 1.
```

```
compute R10_5Zone = 0.
```

```
if (zone = 'R10/5') R10_5zone = 1.
```

```
compute RLDPZone = 0.
```

```
if (zone = 'RLDP') RLDPZone = 1.
```

```
compute RTZone = 0.
```

```
if (zone = 'RT') RTZone = 1.
```

```
compute RSZone = 0.
```

```
if (zone = 'RS') RSZone = 1.
```

MEANS TABLES=sppa BY nbhd /CELLS=MEAN MEDIAN COUNT STDDEV.

*Exclude 372S.

*First 10 Acres.

compute NBHD_10_151 = 0.
compute NBHD_10_152 = 0.
compute NBHD_10_153 = 0.
compute NBHD_10_260 = 0.
compute NBHD_10_261 = 0.
compute NBHD_10_263 = 0.
compute NBHD_10_370 = 0.
compute NBHD_10_371G = 0.
compute NBHD_10_371R = 0.
compute NBHD_10_371S = 0.
compute NBHD_10_372 = 0.
compute NBHD_10_372F = 0.
compute NBHD_10_372N = 0.
compute NBHD_10_372S = 0.
compute NBHD_10_481 = 0.
compute NBHD_10_482 = 0.
compute NBHD_10_483 = 0.
compute NBHD_10_484 = 0.
compute NBHD_10_485 = 0.
compute NBHD_10_485N = 0.
compute NBHD_10_485O = 0.

if (nbhd='151') NBHD_10_151 = first10LN.
if (nbhd='152') NBHD_10_152 = first10LN.
if (nbhd='153') NBHD_10_153 = first10LN.
if (nbhd='260') NBHD_10_260 = first10LN.
if (nbhd='261') NBHD_10_261 = first10LN.
if (nbhd='263') NBHD_10_263 = first10LN.
if (nbhd='370') NBHD_10_370 = first10LN.
if (nbhd='371G') NBHD_10_371G = first10LN.
if (nbhd='371R') NBHD_10_371R = first10LN.
if (nbhd='371S') NBHD_10_371S = first10LN.
if (nbhd='372') NBHD_10_372 = first10LN.
if (nbhd='372F') NBHD_10_372F = first10LN.
if (nbhd='372N') NBHD_10_372N = first10LN.
if (nbhd='372S') NBHD_10_372S = first10LN.
if (nbhd='481') NBHD_10_481 = first10LN.
if (nbhd='482') NBHD_10_482 = first10LN.
if (nbhd='483') NBHD_10_483 = first10LN.
if (nbhd='484') NBHD_10_484 = first10LN.
if (nbhd='485') NBHD_10_485 = first10LN.
if (nbhd='485N') NBHD_10_485N = first10LN.
if (nbhd='485O') NBHD_10_485O = first10LN.

*Excess Land Rates.

compute NBHD_151 = 0.
compute NBHD_152 = 0.
compute NBHD_153 = 0.
compute NBHD_260 = 0.

```
compute NBHD_261 = 0.  
compute NBHD_263 = 0.  
compute NBHD_370 = 0.  
compute NBHD_371G = 0.  
compute NBHD_371R = 0.  
compute NBHD_371S = 0.  
compute NBHD_372 = 0.  
compute NBHD_372F = 0.  
compute NBHD_372N = 0.  
compute NBHD_372S = 0.  
compute NBHD_481 = 0.  
compute NBHD_482 = 0.  
compute NBHD_483 = 0.  
compute NBHD_484 = 0.  
compute NBHD_485 = 0.  
compute NBHD_485N = 0.  
compute NBHD_485O = 0.
```

```
if (nbhd='151') NBHD_151 = excess_land_LN.  
if (nbhd='152') NBHD_152 = excess_land_LN.  
if (nbhd='153') NBHD_153 = excess_land_LN.  
if (nbhd='260') NBHD_260 = excess_land_LN.  
if (nbhd='261') NBHD_261 = excess_land_LN.  
if (nbhd='263') NBHD_263 = excess_land_LN.  
if (nbhd='370') NBHD_370 = excess_land_LN.  
if (nbhd='371G') NBHD_371G = excess_land_LN.  
if (nbhd='371R') NBHD_371R = excess_land_LN.  
if (nbhd='371S') NBHD_371S = excess_land_LN.  
if (nbhd='372') NBHD_372 = excess_land_LN.  
if (nbhd='372F') NBHD_372F = excess_land_LN.  
if (nbhd='372N') NBHD_372N = excess_land_LN.  
if (nbhd='372S') NBHD_372S = excess_land_LN.  
if (nbhd='481') NBHD_481 = excess_land_LN.  
if (nbhd='482') NBHD_482 = excess_land_LN.  
if (nbhd='483') NBHD_483 = excess_land_LN.  
if (nbhd='484') NBHD_484 = excess_land_LN.  
if (nbhd='485') NBHD_485 = excess_land_LN.  
if (nbhd='485N') NBHD_485N = excess_land_LN.  
if (nbhd='485O') NBHD_485O = excess_land_LN.
```

```
compute class1_pct = 0.  
if (class_1 > 0) class1_pct = class_1/total_land_acres.  
compute class2_pct = 0.  
if (class_2 > 0) class2_pct = class_2/total_land_acres.  
compute class3_pct = 0.  
if (class_3 > 0) class3_pct = class_3/total_land_acres.  
compute class4_pct = 0.  
if (class_4 > 0) class4_pct = class_4/total_land_acres.  
compute class5_pct = 0.  
if (class_5 > 0) class5_pct = class_5/total_land_acres.  
compute class6_pct = 0.  
if (class_6 > 0) class6_pct = class_6/total_land_acres.  
compute class7_pct = 0.
```

if (class_7 >0) class7_pct = class_7/total_land_acres.
compute classOT_pct = 0.
if (other_land >0) classOT_pct = other_land/total_land_acres.

compute apples = A1S + A2S + A3S+ A1H + A2H + A3H.
compute cherries = CHS + CHH.
compute pears = PEH + PES.
compute stone_fruit = SFH + SFS.

MEANS TABLES=sppa BY flood_plain /CELLS=MEAN MEDIAN COUNT STDDEV.

*Exclude N.

compute flood_plain_factor = 0.
if (flood_plain = 'F') flood_plain_factor = 1.
compute flood_plain_y = 0.
if (flood_plain = 'Y') flood_plain_y = 1.

MEANS TABLES=sppa BY topography /CELLS=MEAN MEDIAN COUNT STDDEV.

*Likely exclude S.

compute topography_l = 0.
if (topography = 'L') topography_l = 1.
compute topography_p = 0.
if (topography = 'P') topography_p = 1.
compute topography_r = 0.
if (topography = 'R') topography_r = 1.
compute topography_s = 0.
if (topography = 'S') topography_s = 1.

MEANS TABLES=sppa BY gas /CELLS=MEAN MEDIAN COUNT STDDEV.

*Inconclusive on AG, DO NOT INCLUDE.

compute gas_factor = 1.
if(gas = 'N') gas_factor = 0.

MEANS TABLES=sppa BY street_finsh /CELLS=MEAN MEDIAN COUNT STDDEV.

*Probably exclude N.

compute street_finish_d = 0.
if (street_finsh = 'D') street_finish_d = 1.
compute street_finish_n = 0.
if (street_finsh = 'N') street_finish_n = 1.
compute street_finish_g = 0.
if (street_finsh = 'G') street_finish_g = 1.
compute street_finish_p = 0.
if (street_finsh = 'P') street_finish_p = 1.

MEANS TABLES=sppa BY traffic /CELLS=MEAN MEDIAN COUNT STDDEV.

*Probably exclude L.

compute traffic_h = 0.
if (traffic = 'H') traffic_h = 1.
compute traffic_m = 0.
if (traffic = 'M') traffic_m = 1.
compute traffic_n = 0.
if (traffic = 'N') traffic_n = 1.

MEANS TABLES=sppa BY curbs /CELLS=MEAN MEDIAN COUNT STDDEV.

*Inconclusive for AG Land.

compute curb_factor = 1.

if (curbs = 'N') curb_factor = 0.

MEANS TABLES=sppa BY side_walk /CELLS=MEAN MEDIAN COUNT STDDEV.

*Inconclusive for AG Land.

compute side_walk_factor = 1.

if (side_walk = 'N') side_walk_factor = 0.

MEANS TABLES=sppa BY water_shed /CELLS=MEAN MEDIAN COUNT STDDEV.

compute closed_cowiche = 0.

if (water_shed = 'CLOSED - Cowiche Creek #4') closed_cowiche = 1.

compute closed_wenas = 0.

if (water_shed = 'CLOSED - Wenas Creek #1') closed_wenas = 1.

compute closed_ahtanum = 0.

if (water_shed = 'CLOSED - Wide Hollow-Ahtanum #1') closed_ahtanum = 1.

MEANS TABLES=sppa BY fire_risk /CELLS=MEAN MEDIAN COUNT STDDEV.

compute extreme_fire_risk = 0.

if (fire_risk = 'Extreme') extreme_fire_risk = 1.

compute high_fire_risk = 0.

if (fire_risk = 'High') high_fire_risk = 1.

compute moderate_fire_risk = 0.

if (fire_risk = 'Moderate') moderate_fire_risk = 1.

filter off.

*Full Value Model Test BACKWARD.

REGRESSION

/MISSING LISTWISE

/STATISTICS COEFF OUTS R ANOVA

/CRITERIA=PIN(.05) POUT(.10) TOLERANCE(.0001)

/NOORIGIN

/DEPENDENT adj_sale_price

/METHOD=backward apples cherries HOP IRR JGR pears WGR

class_1 class_2 class_3 class_4 class_5 class_6 class_7 other_land

AG_USE RES_USE OT_USE LNAcres excess_land_LN first10LN

topography_p topography_s street_finish_d street_finish_n street_finish_g street_finish_p

RLDPZone RTZone RSZone unbuildable water_available sewer_available

NBHD_151 NBHD_152 NBHD_153 NBHD_260 NBHD_261 NBHD_263 NBHD_370 NBHD_371G NBHD_371R N

BHD_371S NBHD_372

NBHD_372F NBHD_372N NBHD_481 NBHD_482 NBHD_483 NBHD_484 NBHD_485 NBHD_485N NBHD_485

O

NBHD_10_151 NBHD_10_152 NBHD_10_153 NBHD_10_260 NBHD_10_261 NBHD_10_263 NBHD_10_370 N

BHD_10_371G NBHD_10_371R

NBHD_10_371S NBHD_10_372 NBHD_10_372F NBHD_10_372N NBHD_10_481 NBHD_10_482 NBHD_10_4

83 NBHD_10_484 NBHD_10_485

NBHD_10_485N NBHD_10_485O.

REGRESSION

/MISSING LISTWISE

/STATISTICS COEFF OUTS R ANOVA

/CRITERIA=PIN(.05) POUT(.10) TOLERANCE(.0001)

```

/NOORIGIN
/DEPENDENT adj_sale_price
/METHOD=stepwise apples cherries HOP IRR JGR pears WGR
class_1 class_2 class_3 class_4 class_5 class_6 class_7 other_land
AG_USE RES_USE OT_USE LNAcres excess_land_LN first10LN
topography_p topography_s street_finish_d street_finish_n street_finish_g street_finish_p
RLDPZone RTZone RSZone unbuildable water_available sewer_available
NBHD_151 NBHD_152 NBHD_153 NBHD_260 NBHD_261 NBHD_263 NBHD_370 NBHD_371G NBHD_371R N
BHD_371S NBHD_372
NBHD_372F NBHD_372N NBHD_481 NBHD_482 NBHD_483 NBHD_484 NBHD_485 NBHD_485N NBHD_485
O
NBHD_10_151 NBHD_10_152 NBHD_10_153 NBHD_10_260 NBHD_10_261 NBHD_10_263 NBHD_10_370 N
BHD_10_371G NBHD_10_371R
NBHD_10_371S NBHD_10_372 NBHD_10_372F NBHD_10_372N NBHD_10_481 NBHD_10_482 NBHD_10_4
83 NBHD_10_484 NBHD_10_485
NBHD_10_485N NBHD_10_485O.

```

*Full Value Model Final.

REGRESSION

```

/MISSING LISTWISE
/STATISTICS COEFF OUTS R ANOVA
/CRITERIA=PIN(.05) POUT(.10) TOLERANCE(.0001)
/NOORIGIN
/DEPENDENT adj_sale_price
/METHOD=ENTER LNAcres class_1 class_2 class_3 class_4 class_7 excess_land_LN first10LN other_land
AG_USE OT_USE RES_USE RLDPZone street_finish_d street_finish_g street_finish_n topography_p topography
_s
unbuildable water_available apples cherries IRR JGR pears WGR
NBHD_10_261 NBHD_10_370 NBHD_10_371G NBHD_10_371R NBHD_10_372F NBHD_10_372N NBHD_10_
481 NBHD_10_482 NBHD_10_484 NBHD_10_485 NBHD_10_485O
NBHD_370 NBHD_371G NBHD_371R NBHD_371S NBHD_372 NBHD_372F NBHD_372N NBHD_482 NBHD_4
83 NBHD_484 NBHD_485 NBHD_485N
/SAVE PRED (adj_sale_price_pred).

```

compute flat_predicted = adj_sale_price_pred + detached_value.

compute predicted_ratio = flat_predicted / tasp.

compute flat_filter = 1.

if (flat_predicted/tasp < 0.25) flat_filter = 0.

if (flat_predicted/tasp > 1.75) flat_filter = 0.

filter by flat_filter.

RATIO STATISTICS flat_predicted WITH tasp/PRINT=COD MEAN MEDIAN WGTMEAN N.

RATIO STATISTICS flat_predicted WITH tasp by zone/PRINT=COD MEAN MEDIAN WGTMEAN N.

* Market Adjustments.

if (zone='AG') flat_predicted = flat_predicted * 1.072.

if (zone='R10/5') flat_predicted = flat_predicted * 1.047.

if (zone='RLDP') flat_predicted = flat_predicted * 1.25.

if (zone='RS') flat_predicted = flat_predicted * 2.016.

if (zone='RT') flat_predicted = flat_predicted * 1.163.

RATIO STATISTICS flat_predicted WITH tasp by zone/PRINT=COD MEAN MEDIAN WGTMEAN N.

RATIO STATISTICS flat_predicted WITH tasp by acre_range/PRINT=COD MEAN MEDIAN WGTMEAN N.

if (acre_range=0) flat_predicted = flat_predicted * 0.919.

if (acre_range=1) flat_predicted = flat_predicted * 0.997.

if (acre_range=2) flat_predicted = flat_predicted * 1.012.

if (acre_range=3) flat_predicted = flat_predicted * 1.024.

if (acre_range=4) flat_predicted = flat_predicted * 1.035.

if (acre_range=5) flat_predicted = flat_predicted * 0.963.

if (acre_range=6) flat_predicted = flat_predicted * 0.969.

RATIO STATISTICS flat_predicted WITH tasp by acre_range/PRINT=COD MEAN MEDIAN WGTMEAN N.

RATIO STATISTICS flat_predicted WITH tasp BY nbhd (ASCENDING)/PRINT=COD MEAN MEDIAN WGTMEAN N.

if (nbhd='151') flat_predicted = flat_predicted * 0.949.

if (nbhd='152') flat_predicted = flat_predicted * 1.285.

if (nbhd='153') flat_predicted = flat_predicted * 1.024.

if (nbhd='260') flat_predicted = flat_predicted * 0.944.

if (nbhd='261') flat_predicted = flat_predicted * 0.973.

if (nbhd='262') flat_predicted = flat_predicted * 0.964.

if (nbhd='263') flat_predicted = flat_predicted * 0.951.

if (nbhd='370') flat_predicted = flat_predicted * 1.034.

if (nbhd='371G') flat_predicted = flat_predicted * 1.011.

if (nbhd='371R') flat_predicted = flat_predicted * 1.081.

if (nbhd='371S') flat_predicted = flat_predicted * 0.911.

if (nbhd='372') flat_predicted = flat_predicted * 1.057.

if (nbhd='372F') flat_predicted = flat_predicted * 1.355.

if (nbhd='372N') flat_predicted = flat_predicted * 1.033.

if (nbhd='372S') flat_predicted = flat_predicted * 0.85.

if (nbhd='481') flat_predicted = flat_predicted * 0.892.

if (nbhd='482') flat_predicted = flat_predicted * 0.891.

if (nbhd='483') flat_predicted = flat_predicted * 0.948.

if (nbhd='484') flat_predicted = flat_predicted * 1.008.

if (nbhd='485') flat_predicted = flat_predicted * 1.003.

if (nbhd='485N') flat_predicted = flat_predicted * 0.821.

if (nbhd='485O') flat_predicted = flat_predicted * 1.064.

if (nbhd='485P') flat_predicted = flat_predicted * 0.855.

RATIO STATISTICS flat_predicted WITH tasp BY nbhd (ASCENDING)/PRINT=COD MEAN MEDIAN WGTMEAN N.

RATIO STATISTICS flat_predicted WITH tasp/PRINT=COD MEAN MEDIAN WGTMEAN N.

compute predicted_ratio = flat_predicted / tasp.

filter off.

*Test Model 2.

REGRESSION

/MISSING LISTWISE

/STATISTICS COEFF OUTS R ANOVA

/CRITERIA=PIN(.05) POUT(.10) TOLERANCE(.0001)

/NOORIGIN

/DEPENDENT LNspga

/METHOD=STEPWISE apples cherries HOP IRR JGR pears WGR

class1_pct class2_pct class3_pct class4_pct class5_pct class_7 classOT_pct first10LN excess_land_LN LNAcr
es

topography_p topography_s street_finish_d street_finish_n street_finish_g street_finish_p

```

RLDPZone RTZone RSZone unbuildable water_available sewer_available
NBHD_151 NBHD_152 NBHD_153 NBHD_260 NBHD_261 NBHD_263 NBHD_370 NBHD_371G NBHD_371R N
BHD_371S NBHD_372
NBHD_372F NBHD_372N NBHD_481 NBHD_482 NBHD_483 NBHD_484 NBHD_485 NBHD_485N NBHD_485
O
NBHD_10_151 NBHD_10_152 NBHD_10_153 NBHD_10_260 NBHD_10_261 NBHD_10_263 NBHD_10_370 N
BHD_10_371G NBHD_10_371R
NBHD_10_371S NBHD_10_372 NBHD_10_372F NBHD_10_372N NBHD_10_481 NBHD_10_482 NBHD_10_4
83 NBHD_10_484 NBHD_10_485
NBHD_10_485N NBHD_10_485O.

```

*Test Model 2 - Backward.

REGRESSION

/MISSING LISTWISE

/STATISTICS COEFF OUTS R ANOVA

/CRITERIA=PIN(.05) POUT(.10) TOLERANCE(.0001)

/NOORIGIN

/DEPENDENT LNsppa

/METHOD=BACKWARD apples cherries HOP IRR JGR pears WGR

class1_pct class2_pct class3_pct class4_pct class5_pct class_7 classOT_pct first10LN excess_land_LN LNAcr
es

```

topography_p topography_s street_finish_d street_finish_n street_finish_g street_finish_p
RLDPZone RTZone RSZone unbuildable water_available sewer_available
NBHD_151 NBHD_152 NBHD_153 NBHD_260 NBHD_261 NBHD_263 NBHD_370 NBHD_371G NBHD_371R N
BHD_371S NBHD_372
NBHD_372F NBHD_372N NBHD_481 NBHD_482 NBHD_483 NBHD_484 NBHD_485 NBHD_485N NBHD_485
O
NBHD_10_151 NBHD_10_152 NBHD_10_153 NBHD_10_260 NBHD_10_261 NBHD_10_263 NBHD_10_370 N
BHD_10_371G NBHD_10_371R
NBHD_10_371S NBHD_10_372 NBHD_10_372F NBHD_10_372N NBHD_10_481 NBHD_10_482 NBHD_10_4
83 NBHD_10_484 NBHD_10_485
NBHD_10_485N NBHD_10_485O.

```

*Final Model Natural Log.

REGRESSION

/MISSING LISTWISE

/STATISTICS COEFF OUTS R ANOVA

/CRITERIA=PIN(.05) POUT(.10) TOLERANCE(.0001)

/NOORIGIN

/DEPENDENT LNsppa

/METHOD=ENTER LNAcres class1_pct class4_pct class5_pct apples HOP pears WGR

```

RLDPZone street_finish_g street_finish_p topography_p topography_s unbuildable water_available
NBHD_10_261 NBHD_10_370 NBHD_10_371G NBHD_10_371R NBHD_10_372F NBHD_10_372N NBHD_10_
481 NBHD_10_485 NBHD_10_485N NBHD_10_485O
NBHD_153 NBHD_370 NBHD_371G NBHD_372N NBHD_485
/SAVE PRED (LAND_AND_CROP_pred).

```

```

compute Pred_Value = (exp(LAND_AND_CROP_pred) * useable_acres) + detached_value .
compute Pred_Ratio = Pred_Value / tasp.

```

compute pred_filter = 1.

if (Pred_Value/tasp < 0.25) pred_filter = 0.

if (Pred_Value/tasp > 1.75) pred_filter = 0.

filter by pred_filter.

RATIO STATISTICS Pred_Value WITH tasp/PRINT=COD MEAN MEDIAN WGTMEAN N.

RATIO STATISTICS Pred_Value WITH tasp by zone/PRINT=COD MEAN MEDIAN WGTMEAN N.

* Market Adjustments.

if (zone='AG') Pred_Value = Pred_Value * 1.065.

if (zone='R10/5') Pred_Value = Pred_Value * 0.961.

if (zone='RLDP') Pred_Value = Pred_Value * 0.99.

if (zone='RS') Pred_Value = Pred_Value * 1.181.

if (zone='RT') Pred_Value = Pred_Value * 0.964.

RATIO STATISTICS Pred_Value WITH tasp by zone/PRINT=COD MEAN MEDIAN WGTMEAN N.

RATIO STATISTICS Pred_Value WITH tasp by acre_range/PRINT=COD MEAN MEDIAN WGTMEAN N.

if (acre_range=0) Pred_Value = Pred_Value * 1.072.

if (acre_range=1) Pred_Value = Pred_Value * 0.864.

if (acre_range=2) Pred_Value = Pred_Value * 1.046.

if (acre_range=3) Pred_Value = Pred_Value * 1.034.

if (acre_range=4) Pred_Value = Pred_Value * 0.984.

if (acre_range=5) Pred_Value = Pred_Value * 0.949.

if (acre_range=6) Pred_Value = Pred_Value * 1.024.

RATIO STATISTICS Pred_Value WITH tasp by acre_range/PRINT=COD MEAN MEDIAN WGTMEAN N.

RATIO STATISTICS Pred_Value WITH tasp BY nbhd (ASCENDING)/PRINT=COD MEAN MEDIAN WGTMEAN N.

if (nbhd='151') Pred_Value = Pred_Value * 0.872.

if (nbhd='152') Pred_Value = Pred_Value * 1.188.

if (nbhd='153') Pred_Value = Pred_Value * 0.978.

if (nbhd='260') Pred_Value = Pred_Value * 0.968.

if (nbhd='261') Pred_Value = Pred_Value * 1.038.

if (nbhd='263') Pred_Value = Pred_Value * 1.019.

if (nbhd='370') Pred_Value = Pred_Value * 0.983.

if (nbhd='371G') Pred_Value = Pred_Value * 1.013.

if (nbhd='371R') Pred_Value = Pred_Value * 1.106.

if (nbhd='371S') Pred_Value = Pred_Value * 0.962.

if (nbhd='372') Pred_Value = Pred_Value * 0.99.

if (nbhd='372F') Pred_Value = Pred_Value * 1.152.

if (nbhd='372N') Pred_Value = Pred_Value * 0.991.

if (nbhd='372S') Pred_Value = Pred_Value * 0.955.

if (nbhd='481') Pred_Value = Pred_Value * 1.076.

if (nbhd='482') Pred_Value = Pred_Value * 1.027.

if (nbhd='483') Pred_Value = Pred_Value * 0.903.

if (nbhd='484') Pred_Value = Pred_Value * 1.034.

if (nbhd='485') Pred_Value = Pred_Value * 0.999.

if (nbhd='485N') Pred_Value = Pred_Value * 0.923.

if (nbhd='485O') Pred_Value = Pred_Value * 1.012.

if (nbhd='485P') Pred_Value = Pred_Value * 1.894.

if (nbhd='485P') Pred_Value = Pred_Value * 0.855.

RATIO STATISTICS Pred_Value WITH tasp BY nbhd (ASCENDING)/PRINT=COD MEAN MEDIAN WGTMEAN N.

RATIO STATISTICS Pred_Value WITH tasp/PRINT=COD MEAN MEDIAN WGTMEAN N.

compute Pred_Ratio = Pred_Value / tasp.

filter off.

filter by Ratio_Outlier.

compute LandOnlySP = $tasp - (\text{apples} * 1451) - (\text{cherries} * 11782) - (\text{pears} * 17147) - (\text{IRR} * 3712) - (\text{JGR} * 2468) - (\text{WGR} * 9795)$.

compute LandOnlySPPA = LandOnlySP/useable_acres.

compute LNLandOnlysppa = $\ln(\text{LandOnlySPPA})$.

compute blendedFilter = 1.

if (flat_filter = 0 and pred_filter = 0) blendedFilter = 0.

filter off.

filter by blendedFilter.

*Blended!.

REGRESSION

/MISSING LISTWISE

/STATISTICS COEFF OUTS R ANOVA

/CRITERIA=PIN(.05) POUT(.10) TOLERANCE(.0001)

/NOORIGIN

/DEPENDENT LNLandOnlysppa

/METHOD=STEPWISE class1_pct class2_pct class3_pct class4_pct class7_pct classOT_pct first10LN excess_land_LN LNAcres

topography_p topography_s street_finish_d street_finish_n street_finish_g street_finish_p

RLDPZone RTZone RSZone unbuildable water_available sewer_available

NBHD_151 NBHD_152 NBHD_153 NBHD_260 NBHD_261 NBHD_263 NBHD_370 NBHD_371G NBHD_371R NBHD_371S NBHD_372

NBHD_372F NBHD_372N NBHD_481 NBHD_482 NBHD_483 NBHD_484 NBHD_485 NBHD_485N NBHD_485O

NBHD_10_151 NBHD_10_152 NBHD_10_153 NBHD_10_260 NBHD_10_261 NBHD_10_263 NBHD_10_370 NBHD_10_371G NBHD_10_371R

NBHD_10_371S NBHD_10_372 NBHD_10_372F NBHD_10_372N NBHD_10_481 NBHD_10_482 NBHD_10_483 NBHD_10_484 NBHD_10_485

NBHD_10_485N NBHD_10_485O.

REGRESSION

/MISSING LISTWISE

/STATISTICS COEFF OUTS R ANOVA

/CRITERIA=PIN(.05) POUT(.10) TOLERANCE(.0001)

/NOORIGIN

/DEPENDENT LNLandOnlysppa

/METHOD=BACKWARD class1_pct class2_pct class3_pct class4_pct class7_pct classOT_pct first10LN excess_land_LN LNAcres

topography_p topography_s street_finish_d street_finish_n street_finish_g street_finish_p

RLDPZone RTZone RSZone unbuildable water_available sewer_available

NBHD_151 NBHD_152 NBHD_153 NBHD_260 NBHD_261 NBHD_263 NBHD_370 NBHD_371G NBHD_371R NBHD_371S NBHD_372

NBHD_372F NBHD_372N NBHD_481 NBHD_482 NBHD_483 NBHD_484 NBHD_485 NBHD_485N NBHD_485O

NBHD_10_151 NBHD_10_152 NBHD_10_153 NBHD_10_260 NBHD_10_261 NBHD_10_263 NBHD_10_370 NBHD_10_371G NBHD_10_371R

NBHD_10_371S NBHD_10_372 NBHD_10_372F NBHD_10_372N NBHD_10_481 NBHD_10_482 NBHD_10_483 NBHD_10_484 NBHD_10_485

NBHD_10_485N NBHD_10_485O.

REGRESSION

```
/MISSING LISTWISE
/STATISTICS COEFF OUTS R ANOVA
/CRITERIA=PIN(.05) POUT(.10) TOLERANCE(.0001)
/NOORIGIN
/DEPENDENT LNLandOnlysppa
/METHOD=ENTER LNAcres class1_pct class2_pct class3_pct class4_pct classOT_pct RLDPZone
street_finish_d street_finish_g street_finish_n topography_p topography_s unbuildable water_available
NBHD_10_151 NBHD_10_261 NBHD_10_370 NBHD_10_371G NBHD_10_371R NBHD_10_372N NBHD_10_4
81
NBHD_10_482 NBHD_10_484 NBHD_10_485 NBHD_10_485N NBHD_10_485O
NBHD_151 NBHD_152 NBHD_153 NBHD_370 NBHD_371G NBHD_371R NBHD_372N NBHD_483 NBHD_484
NBHD_485
/SAVE PRED (Blended_pred).
```

```
compute Blended_Value = (exp(Blended_pred) * useable_acres) + detached_value + (apples * 1451) + (cherries
* 11782) + (pears * 17147) + (IRR * 3712) + (JGR * 2468) + (WGR * 9795).
```

```
compute Blended_Ratio = Blended_Value / tasp.
```

```
compute outlier = 0.
```

```
if (Blended_Ratio > 1.75) outlier = 1.
```

```
if (Blended_Ratio < 0.25) outlier = 1.
```

```
if (blendedFilter = 0) outlier = 1.
```

```
compute filter = 1.
```

```
if (outlier = 1) filter = 0.
```

```
filter by filter.
```

```
RATIO STATISTICS Blended_Value WITH tasp/PRINT=COD MEAN MEDIAN WGTMEAN N.
```

```
RATIO STATISTICS Blended_Value WITH tasp by zone/PRINT=COD MEAN MEDIAN WGTMEAN N.
```

```
* Market Adjustments.
```

```
if (zone='AG') Blended_Value = Blended_Value * 1.027.
```

```
if (zone='R10/5') Blended_Value = Blended_Value * 0.94.
```

```
if (zone='RLDP') Blended_Value = Blended_Value * 0.958.
```

```
if (zone='RS') Blended_Value = Blended_Value * 1.167.
```

```
if (zone='RT') Blended_Value = Blended_Value * 0.931.
```

```
RATIO STATISTICS Blended_Value WITH tasp by zone/PRINT=COD MEAN MEDIAN WGTMEAN N.
```

```
RATIO STATISTICS Blended_Value WITH tasp by acre_range/PRINT=COD MEAN MEDIAN WGTMEAN N.
```

```
if (acre_range=0) Blended_Value = Blended_Value * 1.004.
```

```
if (acre_range=1) Blended_Value = Blended_Value * 0.974.
```

```
if (acre_range=2) Blended_Value = Blended_Value * 1.02.
```

```
if (acre_range=3) Blended_Value = Blended_Value * 1.055.
```

```
if (acre_range=4) Blended_Value = Blended_Value * 0.958.
```

```
if (acre_range=5) Blended_Value = Blended_Value * 0.97.
```

```
if (acre_range=6) Blended_Value = Blended_Value * 0.999.
```

```
RATIO STATISTICS Blended_Value WITH tasp by acre_range/PRINT=COD MEAN MEDIAN WGTMEAN N.
```

```
RATIO STATISTICS Blended_Value WITH tasp BY nbhd (ASCENDING)/PRINT=COD MEAN MEDIAN WGTMEAN N
```

```
if (nbhd='151') Blended_Value = Blended_Value * 1.001.
```

```
if (nbhd='152') Blended_Value = Blended_Value * 0.98.
```

```

if (nbhd='153') Blended_Value = Blended_Value * 1.01.
if (nbhd='260') Blended_Value = Blended_Value * 1.018.
if (nbhd='261') Blended_Value = Blended_Value * 0.934.
if (nbhd='262') Blended_Value = Blended_Value * 0.896.
if (nbhd='263') Blended_Value = Blended_Value * 1.024.
if (nbhd='370') Blended_Value = Blended_Value * 0.944.
if (nbhd='371G') Blended_Value = Blended_Value * 1.037.
if (nbhd='371R') Blended_Value = Blended_Value * 1.03.
if (nbhd='371S') Blended_Value = Blended_Value * 1.067.
if (nbhd='372') Blended_Value = Blended_Value * 0.99.
if (nbhd='372F') Blended_Value = Blended_Value * 0.931.
if (nbhd='372N') Blended_Value = Blended_Value * 0.996.
if (nbhd='372S') Blended_Value = Blended_Value * 0.969.
if (nbhd='481') Blended_Value = Blended_Value * 1.019.
if (nbhd='482') Blended_Value = Blended_Value * 1.109.
if (nbhd='483') Blended_Value = Blended_Value * 0.939.
if (nbhd='484') Blended_Value = Blended_Value * 1.038.
if (nbhd='485') Blended_Value = Blended_Value * 1.038.
if (nbhd='485N') Blended_Value = Blended_Value * 0.927.
if (nbhd='485O') Blended_Value = Blended_Value * 1.037.
if (nbhd='485P') Blended_Value = Blended_Value * 1.927.
RATIO STATISTICS Blended_Value WITH tasp BY nbhd (ASCENDING)/PRINT=COD MEAN MEDIAN WGTMEAN N

```

```

RATIO STATISTICS Blended_Value WITH tasp/PRINT=COD MEAN MEDIAN WGTMEAN N.

```

```

RATIO STATISTICS Blended_Value WITH TASP /MISSING=EXCLUDE /PRINT=CIN(95) COD MEAN MEDIAN
STDDEV WGTMEAN N.

```