

Regression Model Checking

February 10, 2015

Biostatistics 208

Model checking for linear regression:

- Linearity
- Normality
- Constant variance
- Influential points
- Covariate overlap

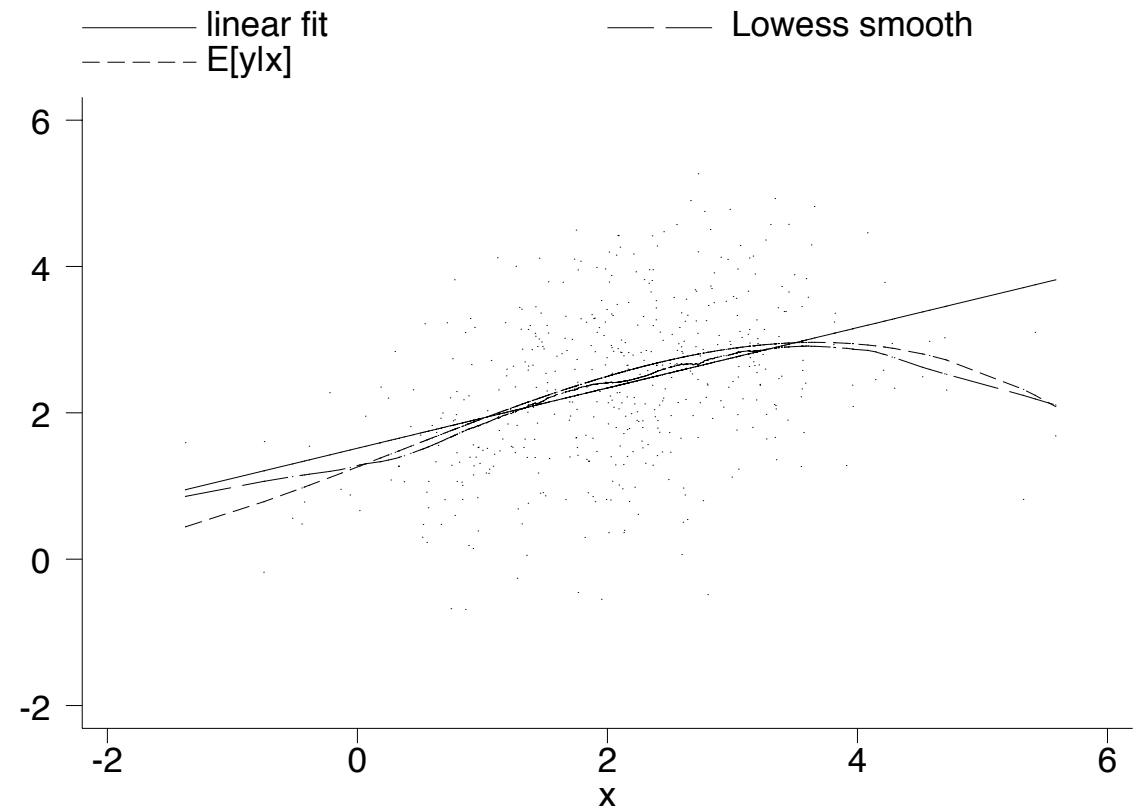
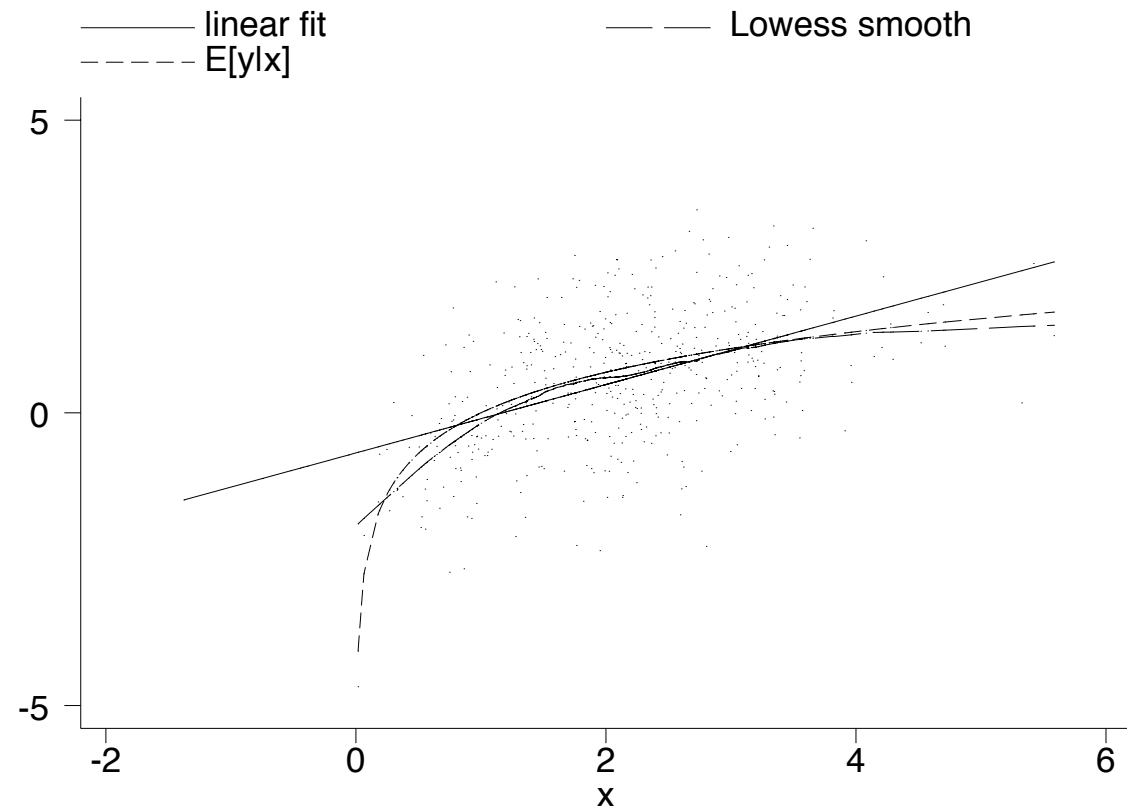
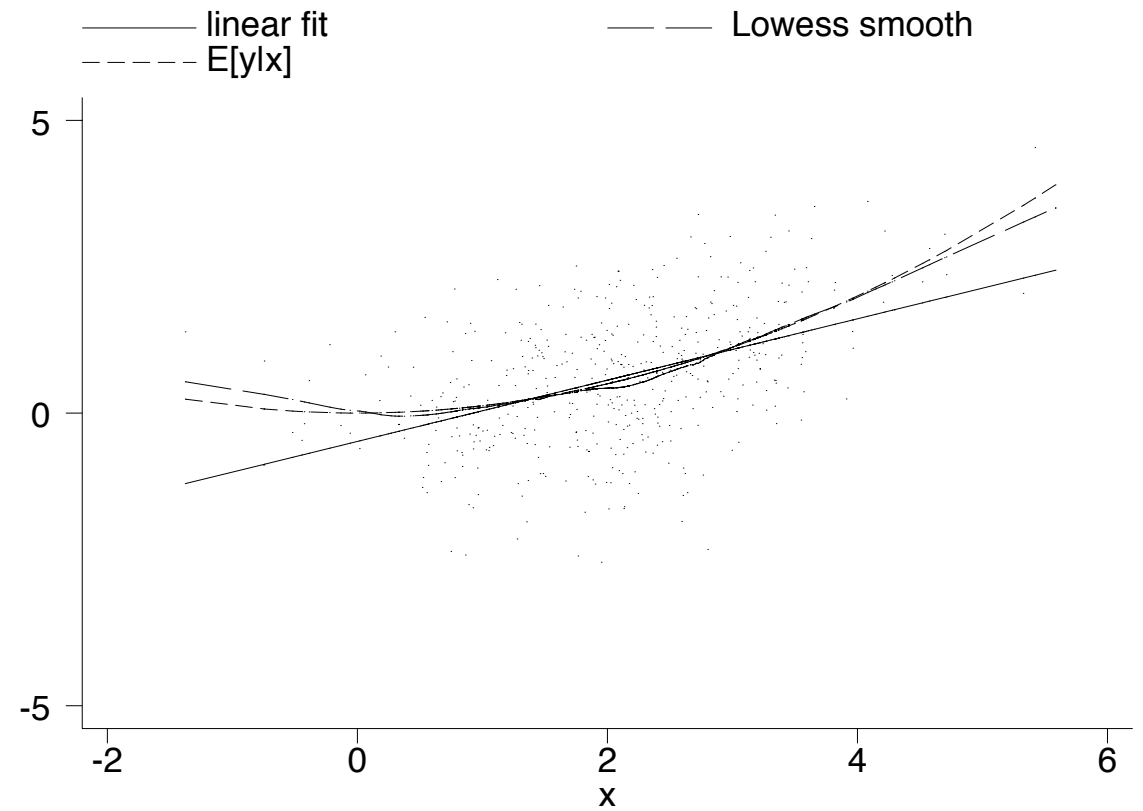
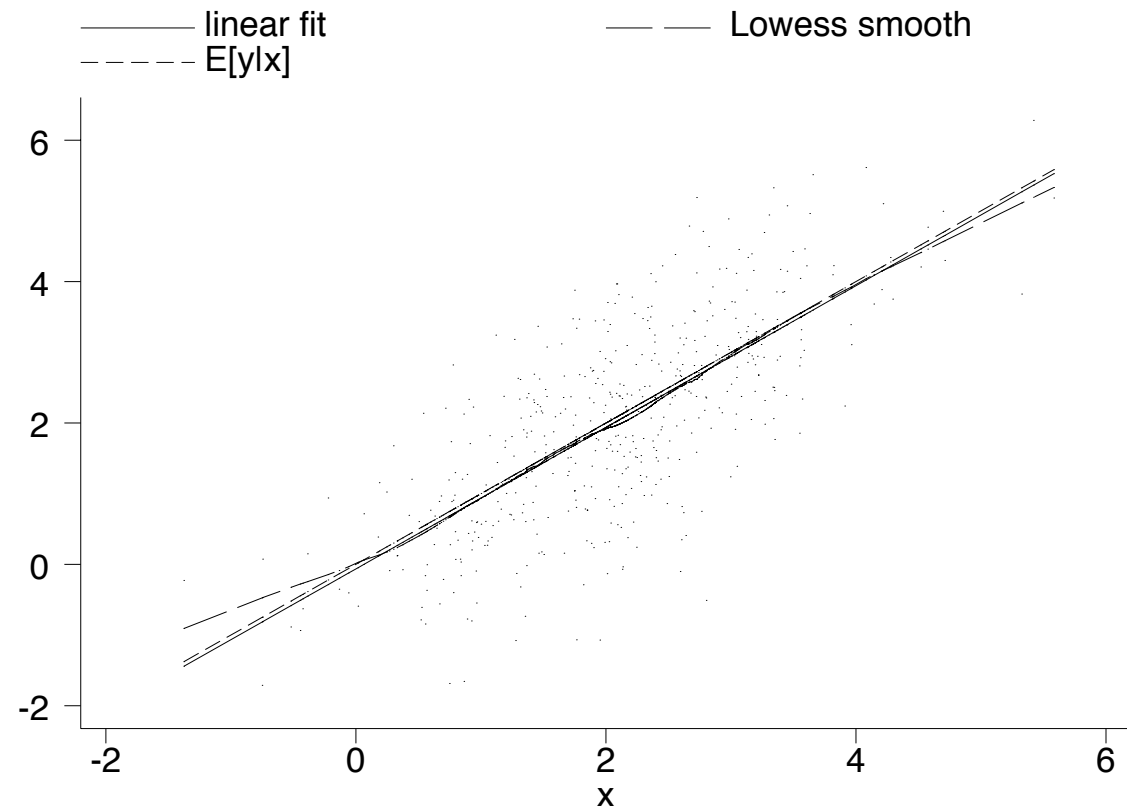
Model checking: linearity

- Regression line assumed linear in continuous predictors
 - i.e., slope is constant
 - equivalently, has no curvature
- If model is correct
 - residuals have mean zero at every value of predictor
 - key to diagnostic tools for departures from linearity

Model checking: linearity

- If assumption badly violated, result can be
 - biased coefficient estimates, residual confounding
 - reduced precision and power, missed real effects
 - misleading, over-simplified conclusions

Three departures from linearity



Diagnostics for nonlinearity: RVP and CPR plots

- To account for effects of other predictors, diagnostics use residuals rather than outcome
- Basic approach: check for non-linear patterns in plots of residuals versus each continuous predictor (RVP) plots
- Better alternative: component plus residual (CPR) plots
 - ─ component due to predictor added back into residual

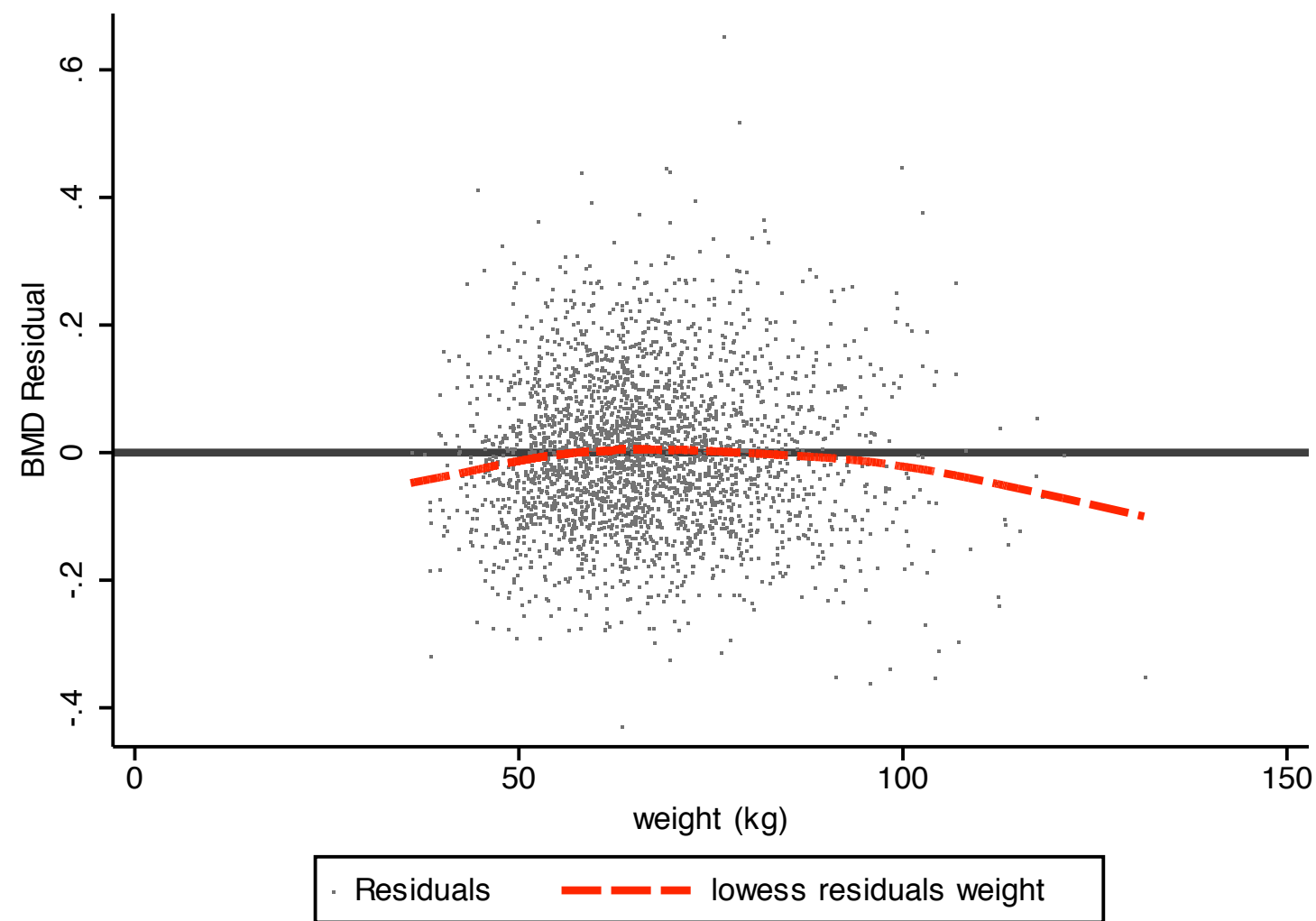
Diagnostics for nonlinearity: RVP and CPR plots

- CPR plots better for diagnosing non-linearity:
 - show trend, RVP plots do not
 - slightly easier to add LOWESS smooth
- Note: use RVP for quadratic, other polynomial models – e.g.,

$$E[y|x] = \beta_0 + \beta_1 x + \beta_2 x^2 + \beta_3 x^3$$

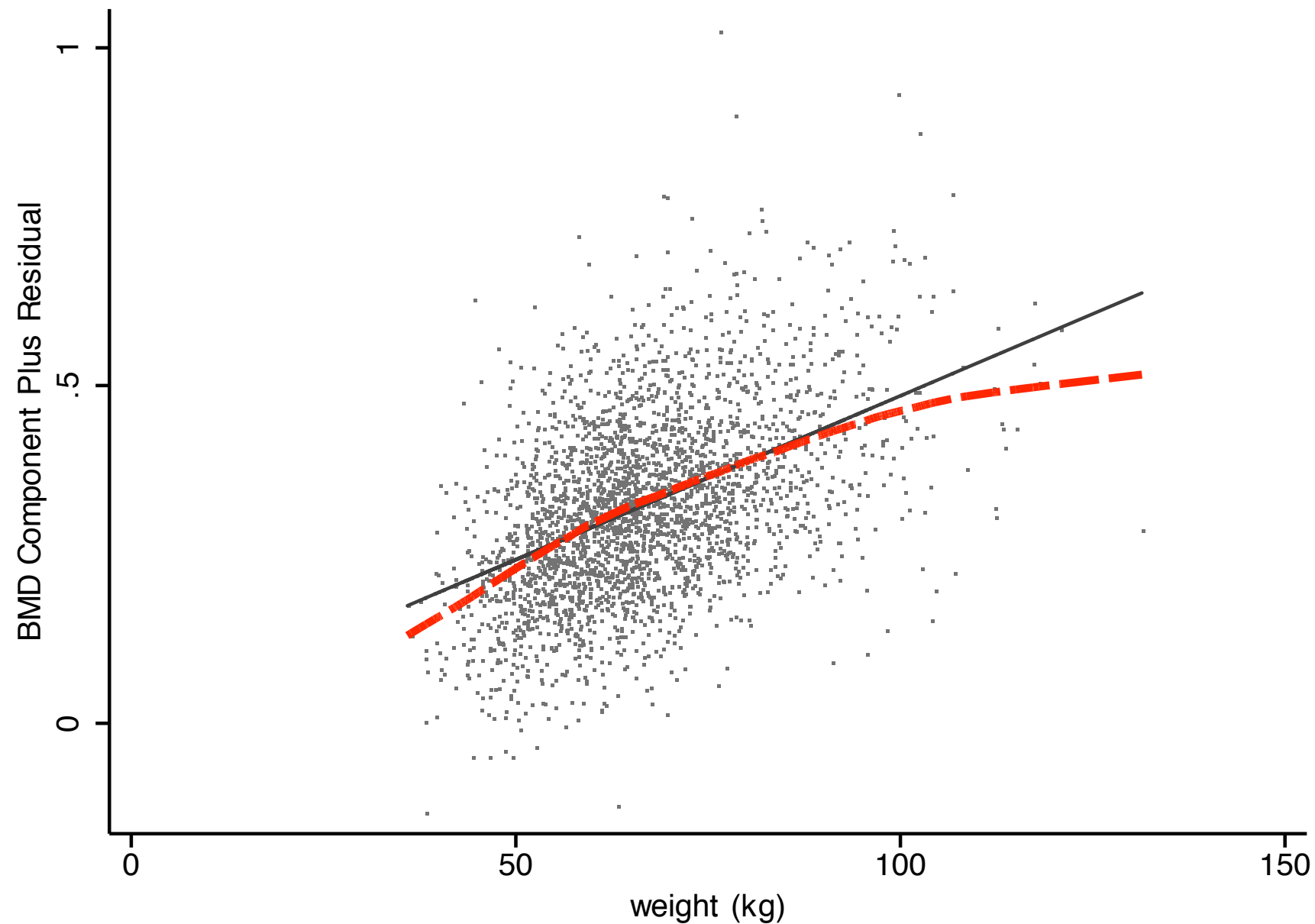
- In both CPR and RVP: mismatch of linear regression line, LOWESS smooth indicates lack of linearity

RVP plot for weight and BMD (SOF data)



Residuals from linear model for dependence of mean BMD on age & weight

CPR plot for weight and BMD



Partial residual for weight = overall residual + estimated linear component from weight

Stata code for RVP and CPR plots

```
regress bmd age weight  
predict residuals, resid
```

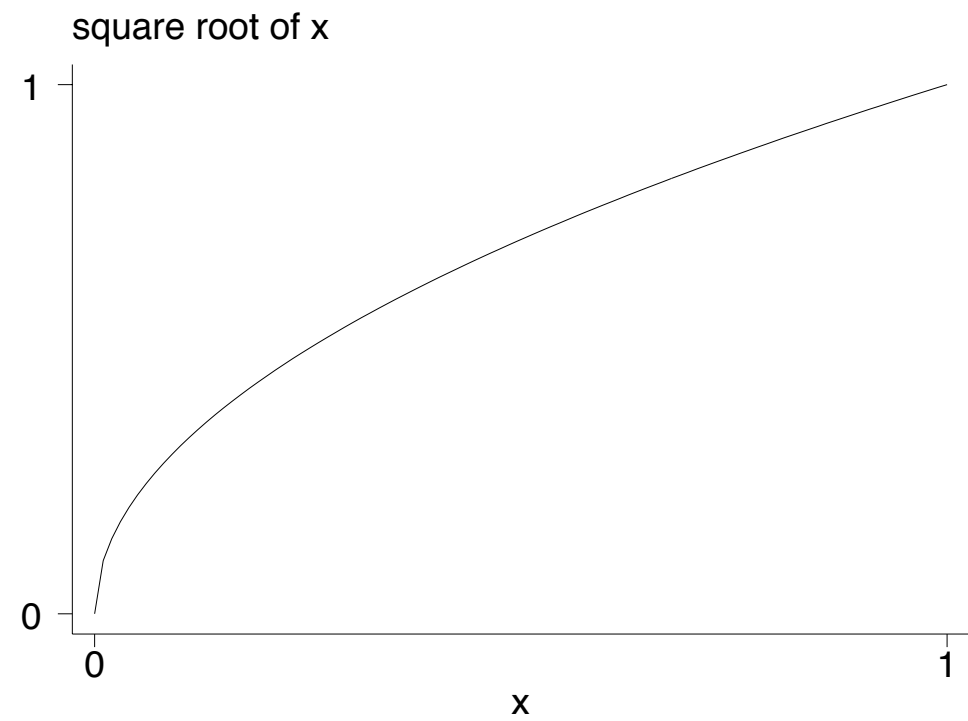
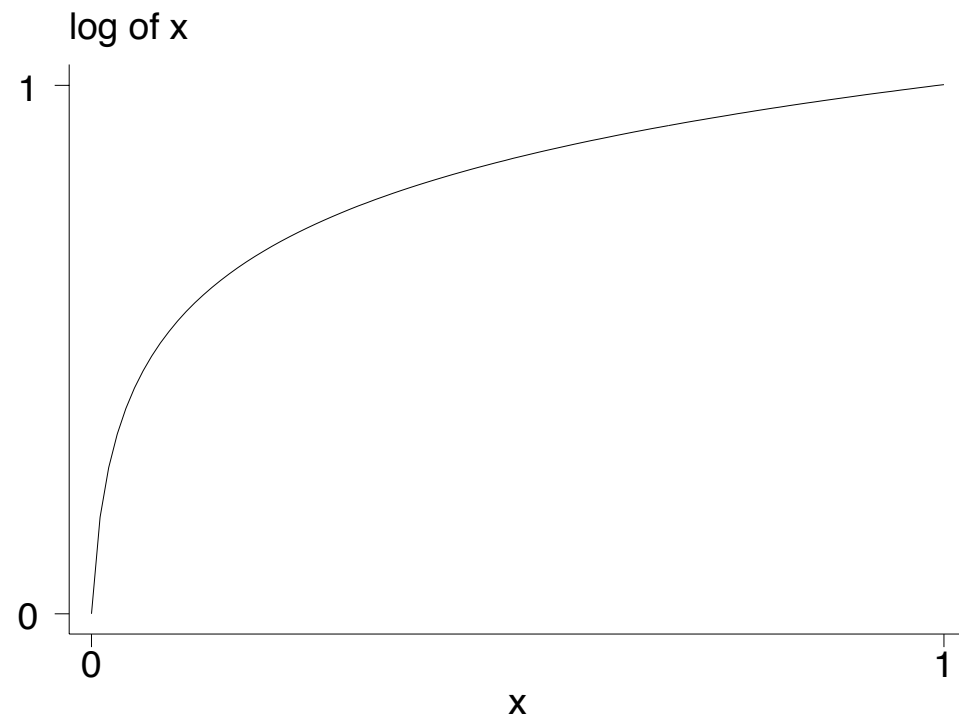
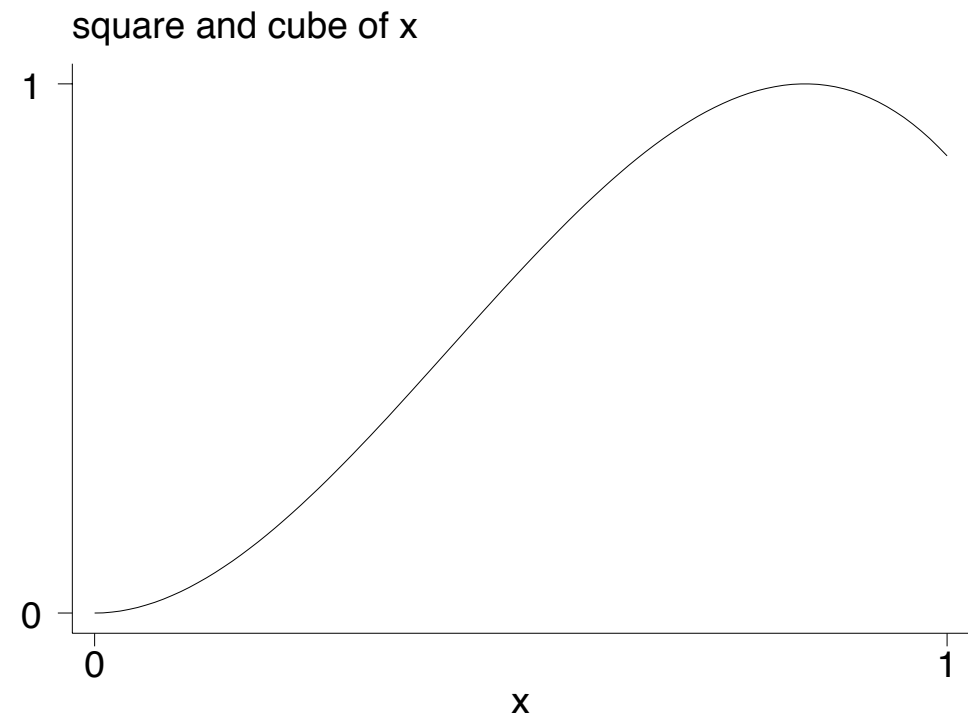
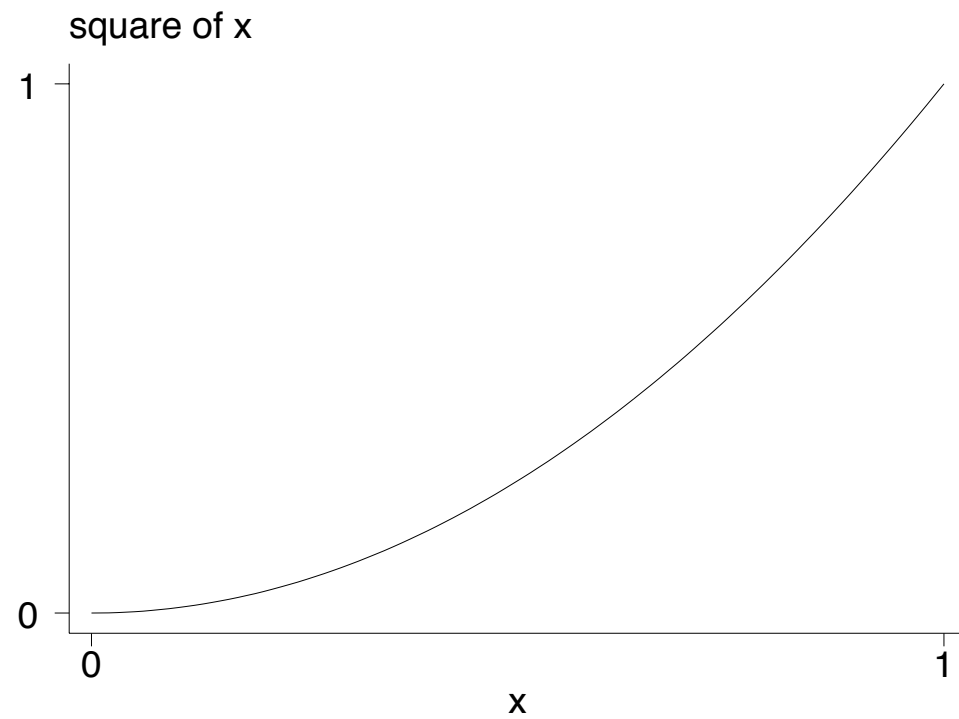
```
rvpplot weight, ///  
    lw(thick) msize(tiny) yline(0, lw(thick)) ///  
    addplot(lowess residuals weight, lw(thick) lcol(red) clpat(longdash)) ///  
    plotregion(style(none)) scheme(slmono) ytitle("BMD Residual") ///  
    xtitle("weight (kg)") xlabel(0(50)150)
```

```
cprplot weight, ///  
    rlopts(clpat(solid) lw(thick)) ///  
    lsopts(bw(.5) clpat(longdash) lw(thick) lcol(red)) ///  
    plotregion(style(none)) scheme(slmono) msize(tiny) ///  
    ytitle("BMD Component Plus Residual") xtitle("weight (kg)") ///  
    xlabel(0(50)150)
```

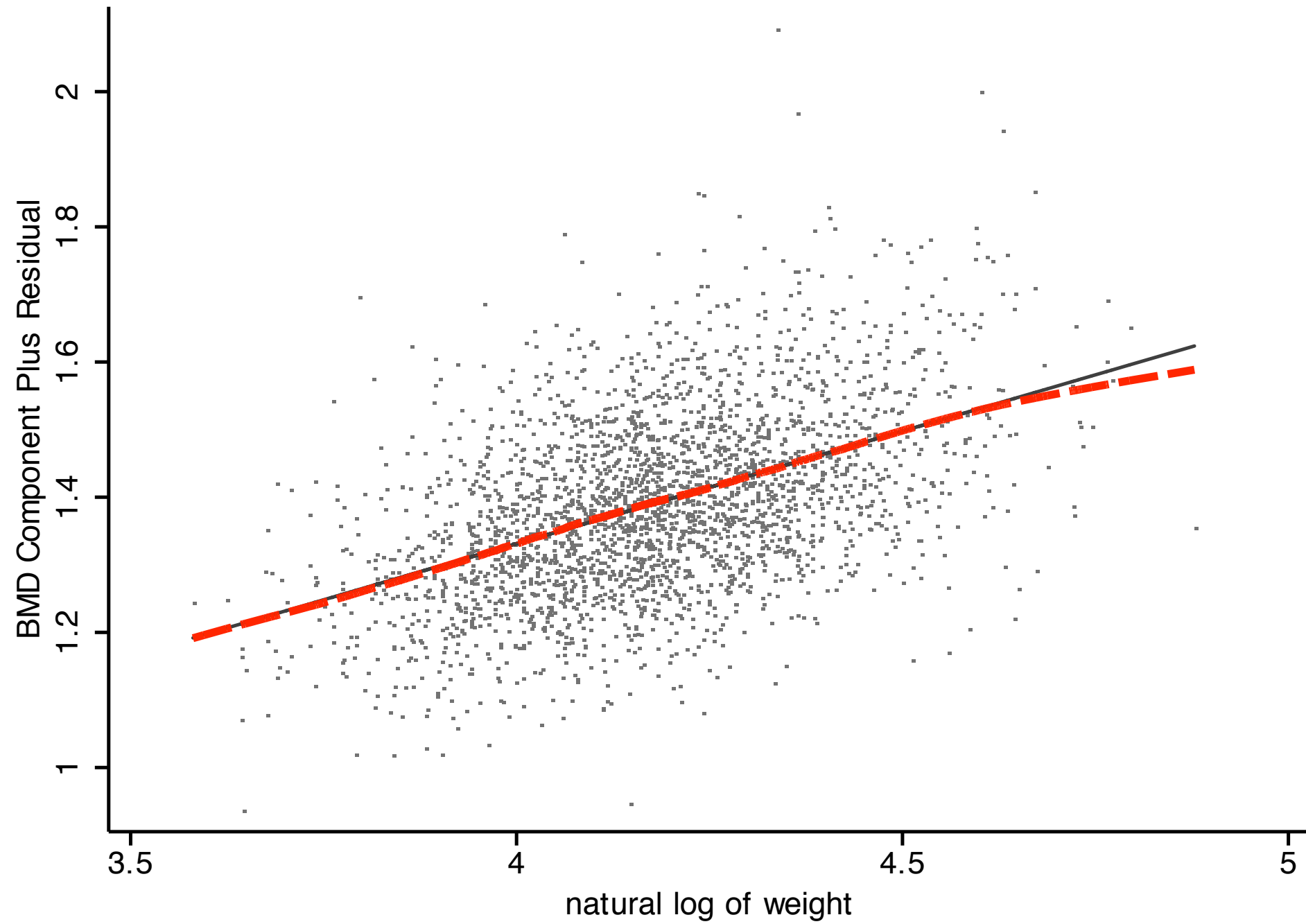
Solution: transform continuous predictors

- Smooth predictor transformations to fix non-linearity:
 - $\log(x)$ – provided $E[y | x]$ is “monotone”
 - square root, cube root, other fractional powers of x
 - x^2, x^3 (lower order terms usually included in the model)

Predictor transformations



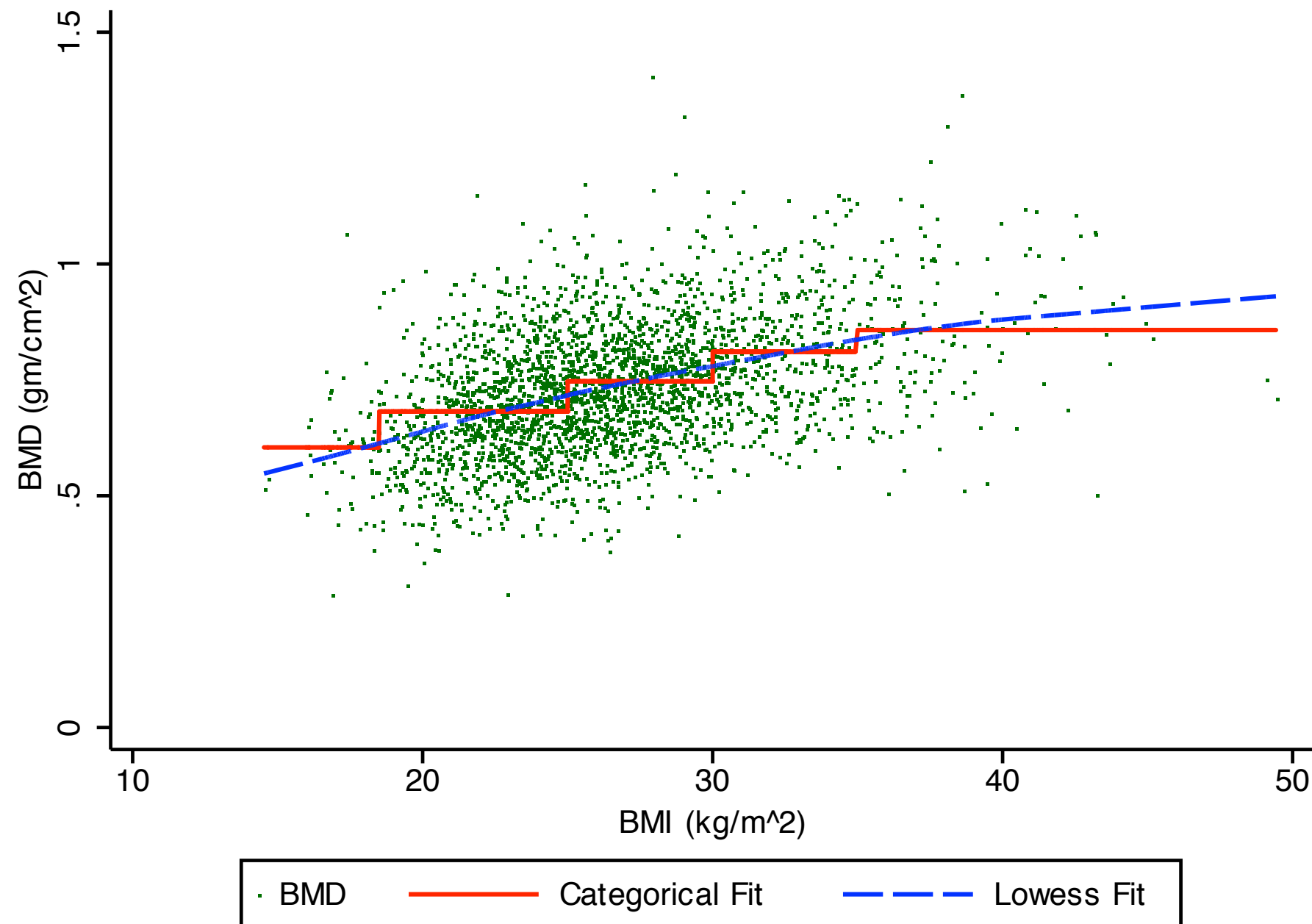
CPR plot for log-weight and BMD (SOF data)



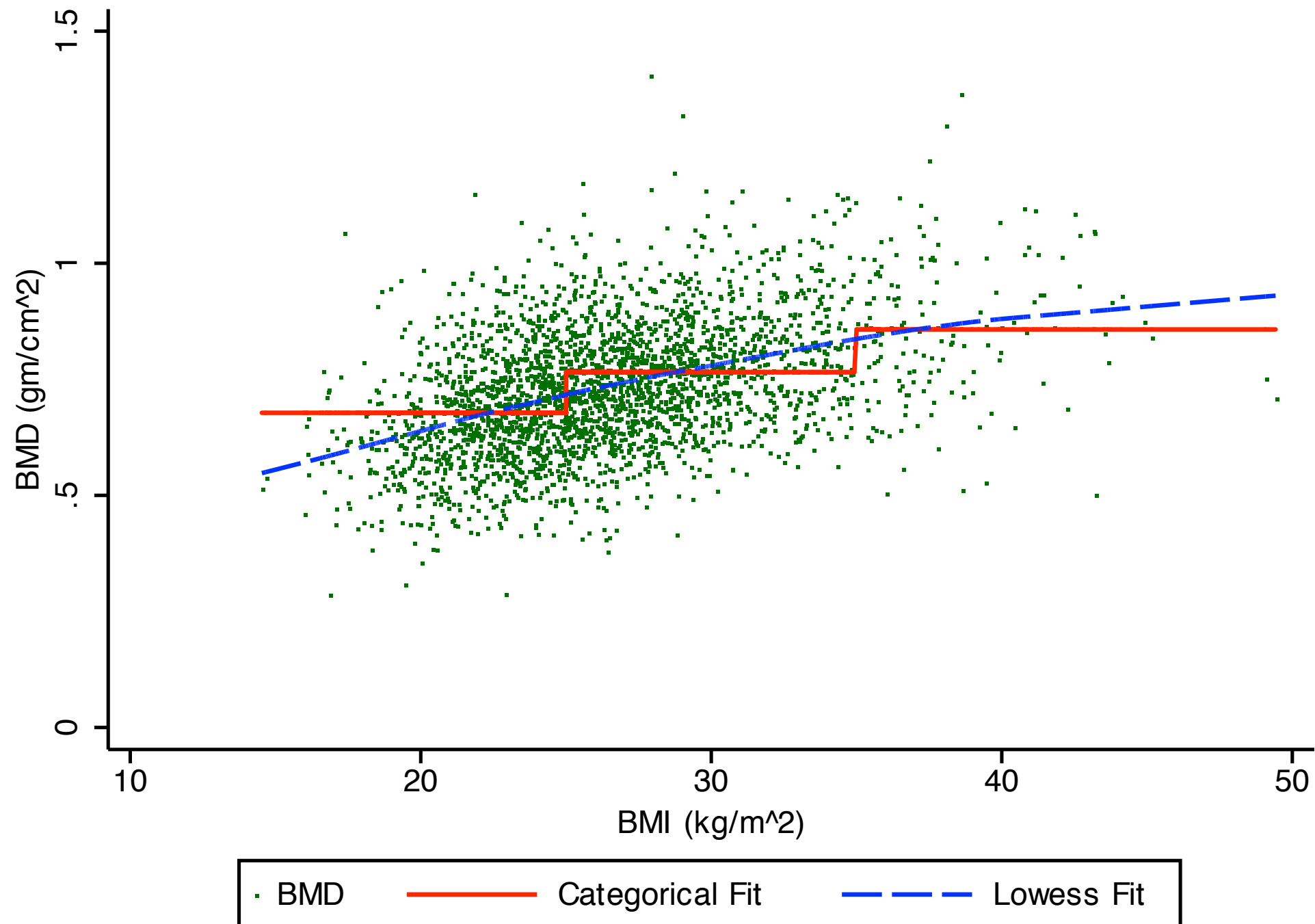
Alternatives: categorize the predictor

- Split at quantiles or clinically familiar cutpoints
- Models mean as a “step function”
- Flexible, familiar, clinically interpretable, but
 - unrealistic if the regression line changes smoothly, sensitive to choice of cut-points, inefficient compared to smooth transformations
- Numbers of categories must balance fit against noisiness

Too coarsely categorizing the predictor



A better trade-off



Stata code for categorical transformations

```
recode bmi min/25=2 25/35=3 35/max=4, gen(bmicat)
regress bmd i.bmicat
predict fitted, xb
twoway ///
    (scatter bmd bmi, msize(vtiny)) ///
    (line fitted bmi, sort lpattern(solid) lcolor(red) lwidth(medthick)) ///
    (lowess bmd bmi, lpattern(longdash) lcolor(blue) lwidth(medthick)), ///
    plotregion(style(none)) scheme(s1color) ///
    ytitle("BMD (gm/cm2)") ///
    xtitle("BMI (kg/m2)") ///
    legend(label(2 "Categorical Fit") label(3 "Lowess Fit") rows(1))
recode bmi min/18.5=1 18.5/25=2 25/30=3 30/35=4 35/max=5, gen(bmicat2)
regress bmd i.bmicat2
predict fitted2, xb
twoway ///
    (scatter bmd bmi, msize(vtiny)) ///
    (line fitted2 bmi, sort lpattern(solid) lcolor(red) lwidth(medthick)) ///
    (lowess bmd bmi, lpattern(longdash) lcolor(blue) lwidth(medthick)), ///
    plotregion(style(none)) scheme(s1color) ///
    ytitle("BMD (gm/cm2)") ///
    xtitle("BMI (kg/m2)") ///
    legend(label(2 "Categorical Fit") label(3 "Lowess Fit") rows(1))
```

Alternatives: linear, restricted cubic splines

- *Linear splines*: piecewise linear with changes in slope at “knots”
 - coefficients interpretable as slope between knots
- *Restricted cubic splines*: smooth curve, cubic between knots
 - tails better behaved than polynomials - *restricted* to linearity before first and after last knot
- easy tests for overall effects and non-linearity, but coefficients uninterpretable; presentation requires plotting
- Also: fractional polynomials (`fracpoly` command)

Linear spline model for BMI effect on BMD in SOF

```
. mkspline bmi1 18.5 bmi2 25 bmi3 30 bmi4 35 bmi5 = bmi
```

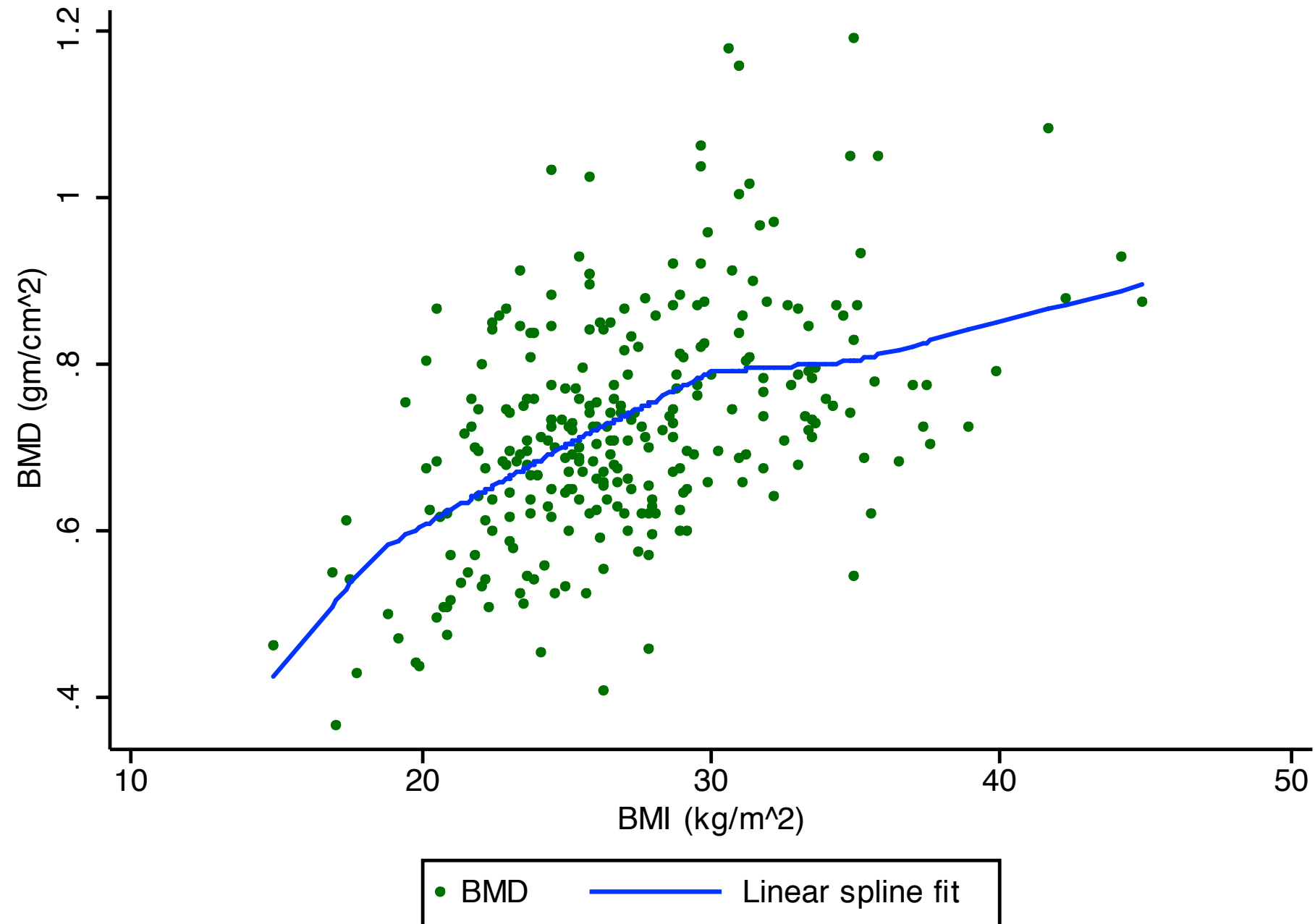
```
. regress bmd bmi1-bmi5
```

Source	SS	df	MS	Number of obs	=	278
Model	1.34269169	5	.268538337	F(5, 272)	=	18.91
Residual	3.86165215	272	.014197251	Prob > F	=	0.0000
				R-squared	=	0.2580
				Adj R-squared	=	0.2444
Total	5.20434383	277	.018788245	Root MSE	=	.11915

bmd	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
bmi1	.0418738	.0300524	1.39	0.165	-.017291	.1010387
bmi2	.0194547	.0060541	3.21	0.001	.0075358	.0313736
bmi3	.017719	.0054267	3.27	0.001	.0070354	.0284027
bmi4	.0024954	.0070065	0.36	0.722	-.0112986	.0162893
bmi5	.0094409	.007597	1.24	0.215	-.0055154	.0243972
_cons	-.1979034	.5417402	-0.37	0.715	-1.26444	.8686334

Interpretation of the overall F statistic for this model?

Linear spline fit



Testing for non-linearity using linear splines

```
. testparm bmi*, equal
```

```
(1) -bmi1+bmi2=0
```

```
(2) -bmi1+bmi3=0
```

```
(3) -bmi1+bmi4=0
```

```
(4) -bmi1+bmi5=0
```

```
F(4, 272)= 2.24
```

```
Prob > F = 0.0654
```

F statistic evaluates equality of slopes between knots.

Cubic spline model allowing for separate trends in viral load, in patients with wild type and drug-resistant HIV

```
. mkspline dursp = duration, cubic knots(30 60 90 180 360)
. forvalues i = 1/4 {
2.     forvalues j = 0/1 {
3.         gen dursp'i'_'j' = dursp'i'*(Anyresistance=='j')
4.     }
5. }
. xtmixed logvl Anyresistance dursp*_0 dursp*_1 || studyid: duration, cov(uns)
```

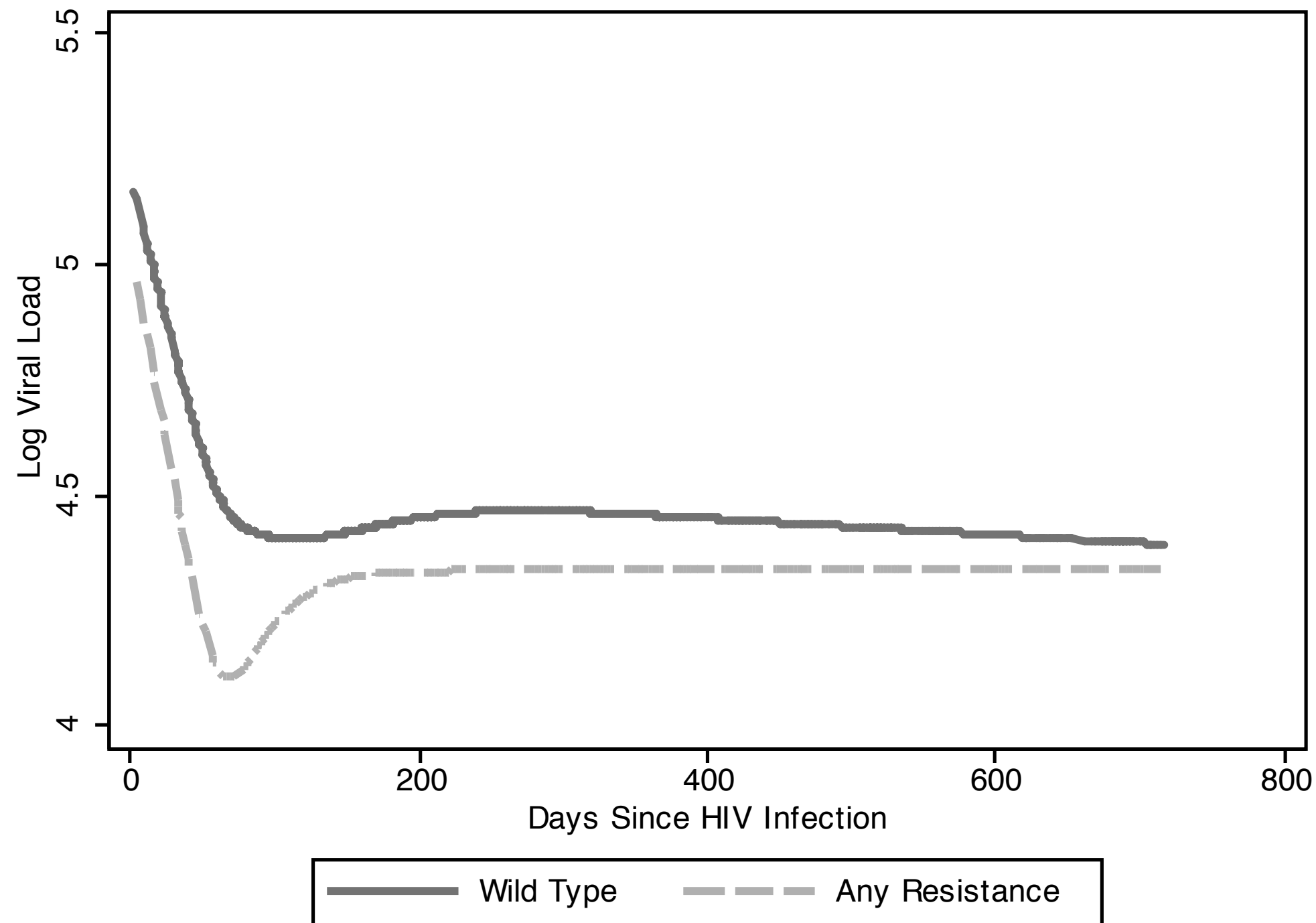
logvl	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]	
Anyresis~e	-.1331557	.279688	-0.48	0.634	-.6813342	.4150227
dursp1_0	-.0121014	.0021882	-5.53	0.000	-.0163902	-.0078127
dursp2_0	.2147899	.0578857	3.71	0.000	.1013359	.3282439
dursp3_0	-.4045134	.1264812	-3.20	0.001	-.6524119	-.1566148
dursp4_0	.1795569	.0734066	2.45	0.014	.0356825	.3234313
dursp1_1	-.0172685	.0046144	-3.74	0.000	-.0263127	-.0082244
dursp2_1	.4717277	.1221695	3.86	0.000	.2322799	.7111754
dursp3_1	-1.002658	.2663056	-3.77	0.000	-1.524607	-.4807085
dursp4_1	.5502881	.1538577	3.58	0.000	.2487326	.8518436
_cons	5.178585	.1207889	42.87	0.000	4.941843	5.415327

wild type

resistant

Linear “mixed effects” model, for log VL, accounting for within-individual dependence in repeated VL measures (mixed effects models covered in Biostat 209).

Cubic spline model for trends in viral load, in patients with



- Test for any time effect on VL in drug resistant group

```
. testparm dursp1_1 dursp2_1 dursp3_1 dursp4_1
( 1)  [logvl]dursp1_1 = 0
( 2)  [logvl]dursp2_1 = 0
( 3)  [logvl]dursp3_1 = 0
( 4)  [logvl]dursp4_1 = 0
      chi2( 4) = 20.54
```

- Test for departure from linearity in drug resistant group

```
. testparm dursp2_1 dursp3_1 dursp4_1
( 1)  [logvl]dursp2_1 = 0
( 2)  [logvl]dursp3_1 = 0
( 3)  [logvl]dursp4_1 = 0
      chi2( 3) = 19.57
      Prob > chi2 = 0.0002
```

- Similar code used for testing within wild type group
- Equivalent testparm syntax for testing overall time effect:

```
testparm dursp1_*
```

- Formal testing for between-group differences is complex

Checking linearity: summary

- Diagnostics:
 - linear models: curved LOWESS smooth in CPR or RVP plot
 - more generally (i.e., linear, logistic, Cox models): fit restricted cubic spline, test for departure from linearity using `testparm` for all but first spline component
- Solutions: transform predictor, use linear or cubic splines

Checking the model: normality

- Validity of t - and F -tests, CIs depend on assumption of normality of residuals (ε)
- Fairly robust to violations, especially short-tailed errors in larger samples
- However, long-tailed distributions of ε can degrade power, precision
- Diagnostics: Q-Q and other plots of residuals

Example: model for mean LDL in HERS

```
. regress LDL BMI age nonwhite smoking drinkany
```

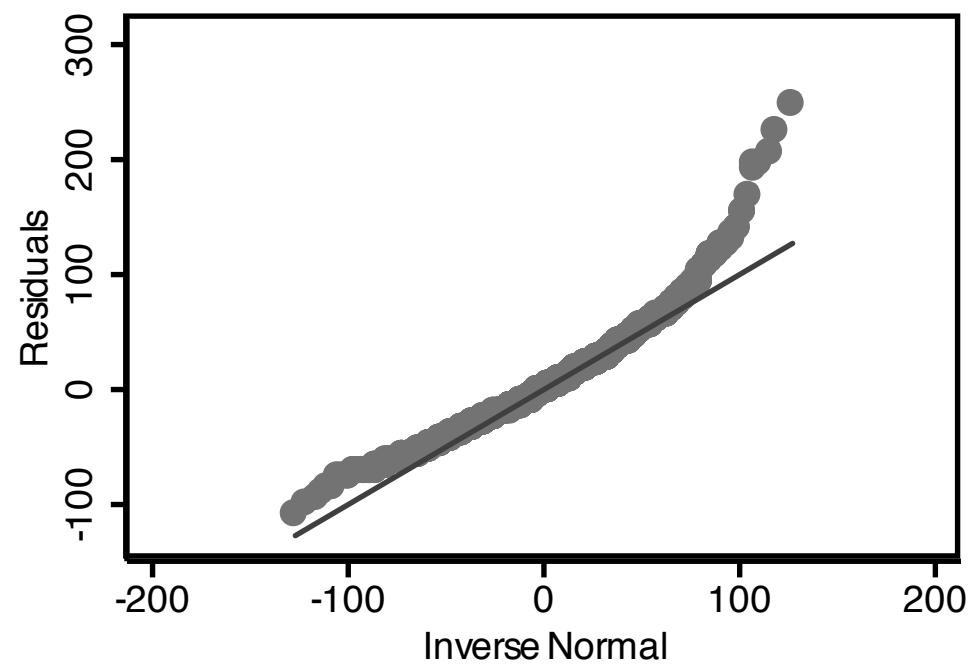
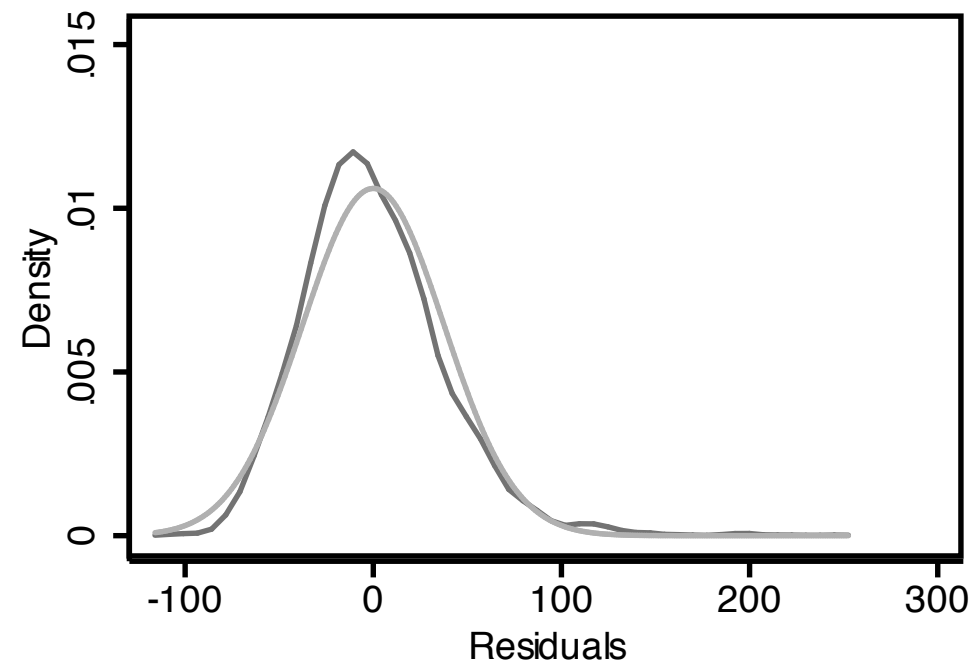
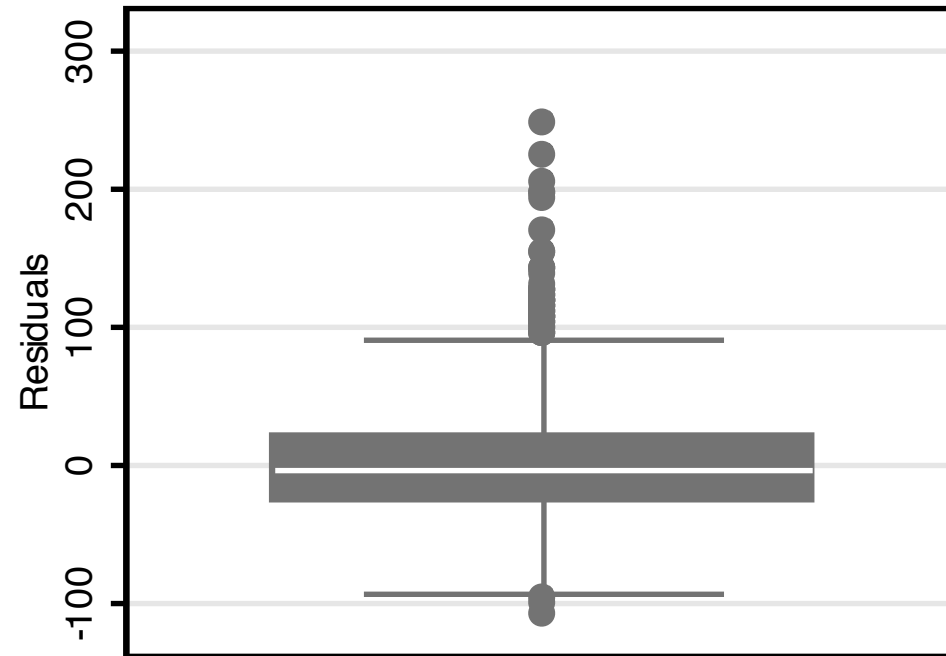
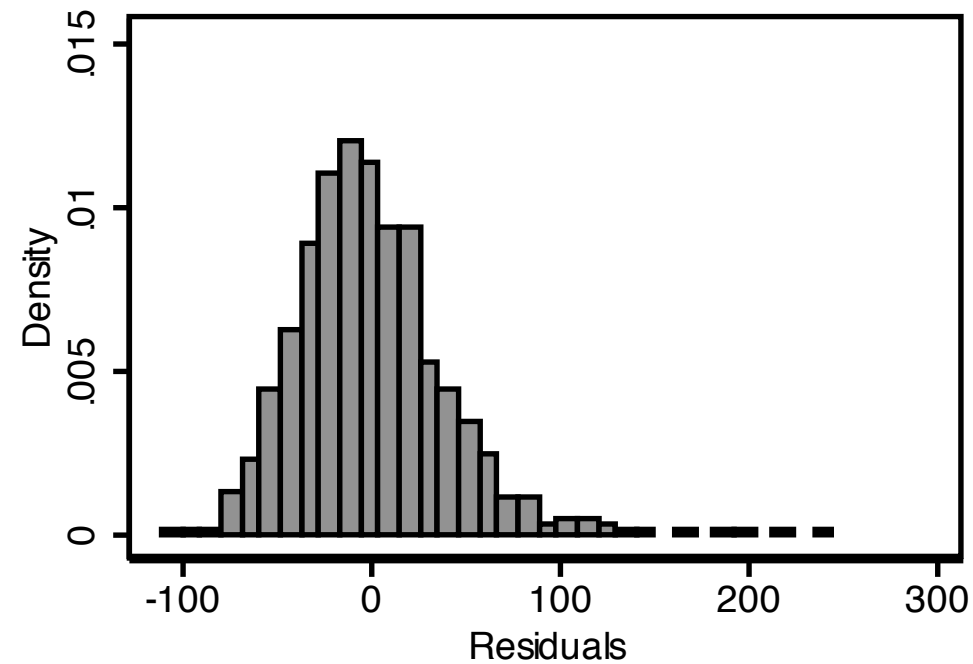
Source	SS	df	MS	Number of obs	=	2745
Model	42279.1877	5	8455.83753	F(5, 2739)	=	5.97
Residual	3881903.3	2739	1417.27028	Prob > F	=	0.0000
				R-squared	=	0.0108
				Adj R-squared	=	0.0090
Total	3924182.49	2744	1430.09566	Root MSE	=	37.647

LDL	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
BMI	.3591038	.1341047	2.68	0.007	.0961472	.6220605
age	-.1897166	.1130776	-1.68	0.094	-.4114427	.0320095
nonwhite	5.219436	2.323673	2.25	0.025	.6631079	9.775764
smoking	4.750738	2.210391	2.15	0.032	.4165362	9.08494
drinkany	-2.722354	1.498854	-1.82	0.069	-5.661352	.2166445
_cons	147.3153	9.256449	15.91	0.000	129.165	165.4656

```
* save residuals in new variable
```

```
. predict resid, resid
```

Diagnosing departures from normality

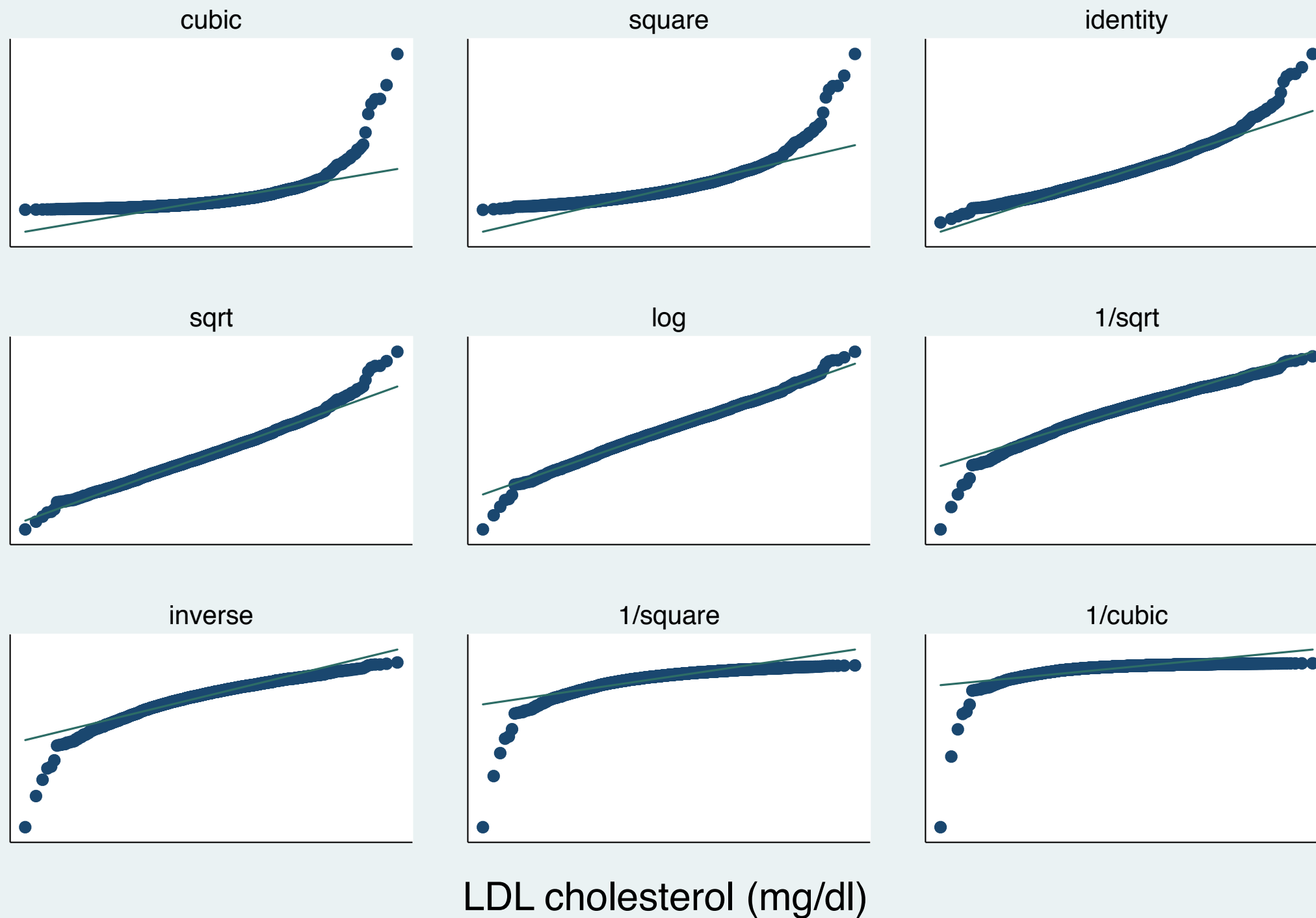


Some apparent evidence for skewness

Solution: transform the outcome

- Residuals skewed (usually to the right):
 - log, square root, other power transformations
 - may need to add constant to make all values positive
- Search for best transformation using `q1adder` command in Stata
- Residuals symmetrically long-tailed
 - rank transformation, trimming, Winsorization

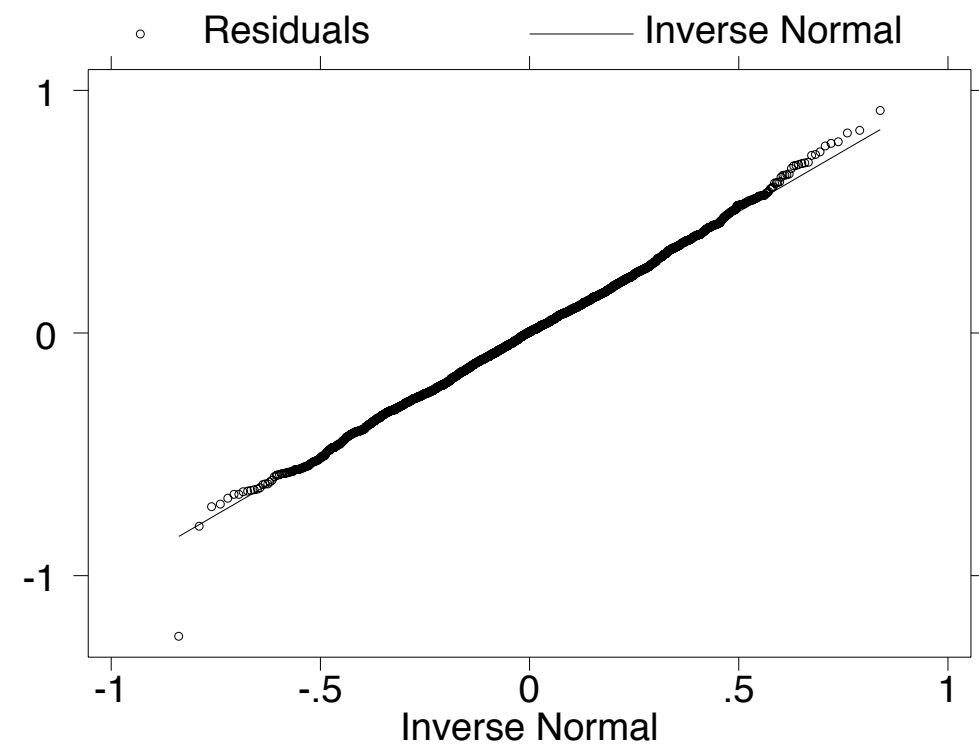
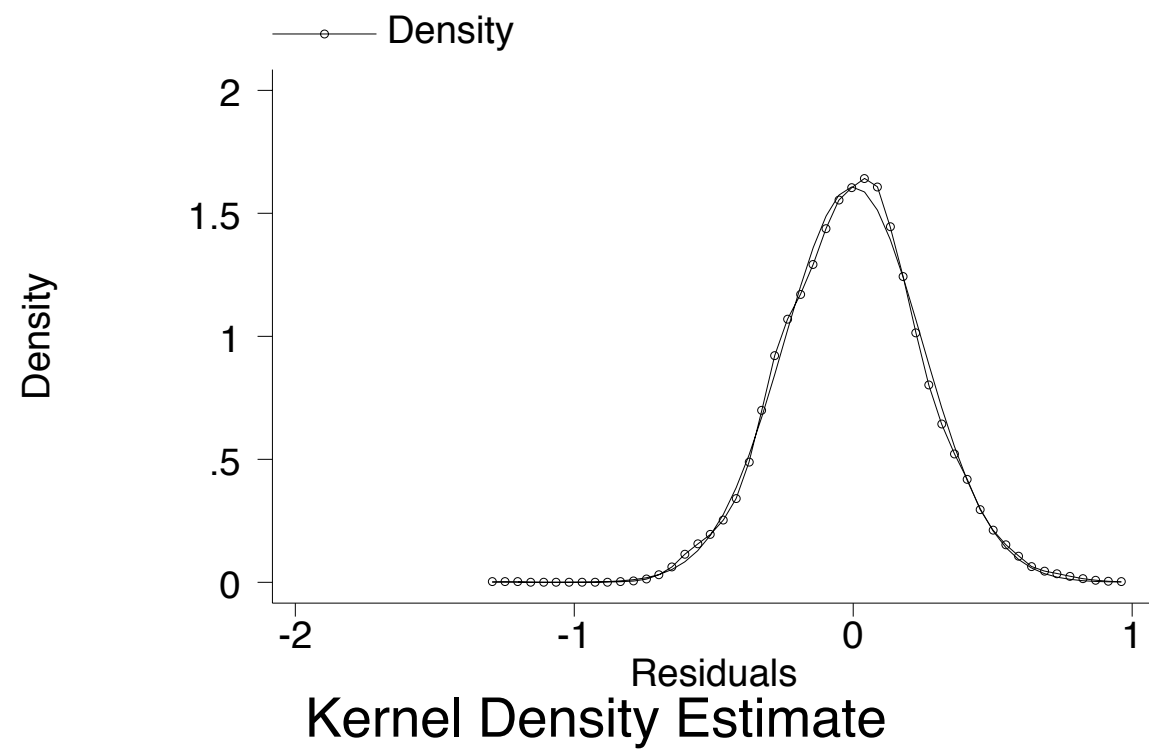
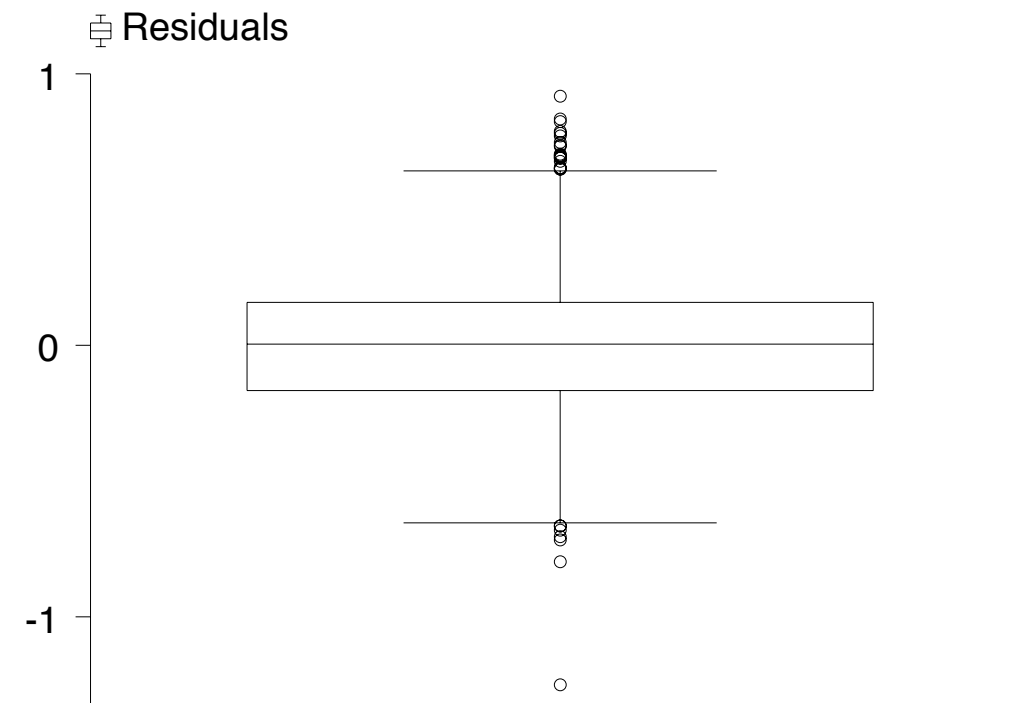
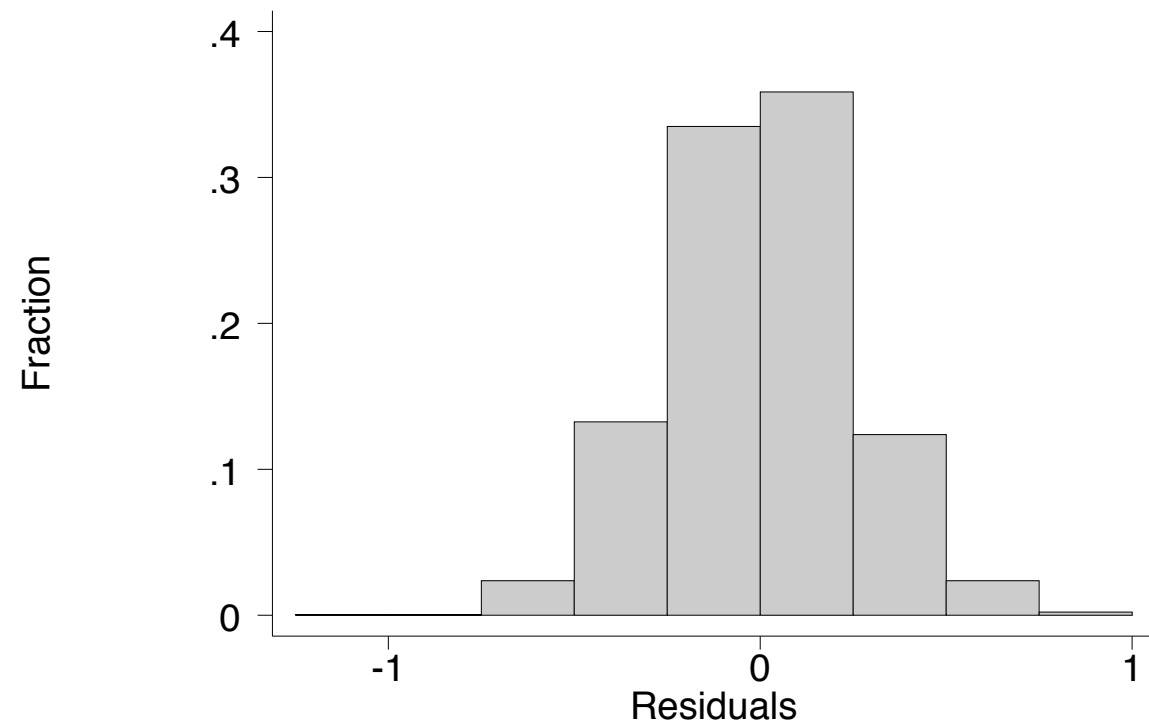
Q ladder plots for LDL



Quantile-Normal plots by transformation

Stata command: `qladder LDL, ylabel(none) xlabel(none)`

Residuals of log-transformed LDL



Another solution: bootstrap CIs

- Resample n observations *with replacement* from data, re-fit model, store estimates, repeat 100, 500, 1,000 times or more
- Distribution of bootstrap estimates models sampling distribution of actual estimate
- Quick, partial solution:
 1. replace model-based SE by SD of bootstrap estimates
 2. construct CIs assuming Normality (i.e. estimate $\pm 1.96 \cdot \text{SE}$)

A better solution: percentile bootstrap CIs

- 95% CI: 2.5th to 97.5th percentile of bootstrap estimates
- Bias-correction shifts CI slightly to right or left
- Slower but avoids making Normality assumption
- Requires using many (≥ 500) bootstrap samples
 - extreme percentiles are noisy!
 - default is 50

Stata code for normal based bootstrap CIs

```
. regress LDL BMI age nonwhite smoking drinkany, vce(bootstrap)
(running regress on estimation sample)
```

```
Bootstrap replications (50)
-----+--- 1 ----+--- 2 ----+--- 3 ----+--- 4 ----+--- 5
..... 50
```

```
Linear regression                Number of obs      =      2745
                                Replications          =       50
                                Wald chi2(5)           =     28.49
                                Prob > chi2            =     0.0000
                                R-squared               =     0.0108
                                Adj R-squared           =     0.0090
                                Root MSE             =    37.6467
```

	LDL	Observed Coef.	Bootstrap Std. Err.	z	P> z	Normal-based [95% Conf. Interval]	
	BMI	.3591038	.1462864	2.45	0.014	.0723878	.6458199
	age	-.1897166	.1020424	-1.86	0.063	-.3897161	.0102829
	nonwhite	5.219436	2.638803	1.98	0.048	.047478	10.39139
	smoking	4.750738	2.186506	2.17	0.030	.465265	9.036211
	drinkany	-2.722354	1.540028	-1.77	0.077	-5.740754	.2960465
	_cons	147.3153	8.558059	17.21	0.000	130.5418	164.0888

Stata code for bias-corrected percentile CIs

```
. quietly regress LDL BMI age nonwhite smoking drinkany, vce(bootstrap, reps(1000))  
. estat bootstrap
```

Linear regression

Number of obs = 2745
Replications = 1000

LDL	Observed Coef.	Bias	Bootstrap Std. Err.	[95% Conf. Interval]		
BMI	.35910384	.0041909	.13067908	.1089804	.6241642	(BC)
age	-.1897166	-.0004125	.11328262	-.3846027	.0482868	(BC)
nonwhite	5.2194359	.0693828	2.5754791	.0820059	10.25661	(BC)
smoking	4.750738	.0231322	2.2627561	.4212856	9.389248	(BC)
drinkany	-2.7223535	.041779	1.5069455	-5.879202	-.0033287	(BC)
_cons	147.31529	-.1740116	9.0724581	129.3511	164.1917	(BC)

(BC) bias-corrected confidence interval

Note:“estat bootstrap, all“ provides normal, percentile & BC intervals

Solution: Generalized Linear Models (GLMs)

- GLMs model a transform of the mean (rather than a transform of the outcome), do not require normality
- Example: logistic model for binary outcomes uses logit transformation of the mean (proportion):

$$\log \frac{E[y|x_1, x_2, \dots]}{1 - E[y|x_1, x_2, \dots]} = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \dots$$

- Other generalized linear models (GLMs) for various (e.g. count and continuous) outcomes, generally model $\log E[y | x]$ as linear (Biostat 209)
 - gamma, Poisson, negative binomial, zero-inflated models

Another solution: ordinal models

- Example: distribution of Agatston scores for coronary artery calcium (CAC) mostly zeroes with long right tail
- Log-transformation (after adding 1) does not help: still mostly zeroes with long right tail
- Could dichotomize outcome as $CAC > 0$ or $CAC > 10$
 - use logistic model - but potentially wasteful

Ordinal models (continued)

- Alternatively, categorize CAC as 0, 1-9, 10-99, 100-399, ≥ 400 , use regression model for ordinal outcomes
- proportional odds (`ologit` - to be discussed with logistic regression)
- Proportional odds assumption relaxed using `gologit2`

Solution: two-part and zero-inflated models

- *Two-part models*:
 1. logistic model for $CAC > 0$
 2. linear model for log-transformed CAC scores > 0
- Testing of overall predictor effects uses *seemingly unrelated regressions* approach, `suest` command
- *Zero-inflated models* allow for more zeroes than expected under Poisson or negative binomial distribution (and can be applied to non-count outcomes); `zip` and `zinb` commands

Checking normality: summary

- Diagnostics: curvature in QQ-plot of residuals
- Solutions: transform outcome, use bootstrap percentile CIs, or GLMs including ordinal, two-part, zero-inflated models

Checking the model: constant variance

- If constant variance assumption is violated
 - coefficient estimates unbiased but inefficient
 - tests for between-group differences may be invalid
 - unlike Normality problems, larger samples don't help

Diagnostics: constant variance

- Plot residuals against fitted values, predictors
 - check for horizontal funnel shapes
- Compare sample size, variance of residuals across subgroups:
 - watch out if both differ by factors of more than 2

Solution: transform outcome

<i>outcome</i>	<i>transformation</i>
variance $\propto$ mean	square root
SD $\propto$ mean	log
proportions	arcsin
correlations	$\log[(1 + \rho)/(1 - \rho)]$

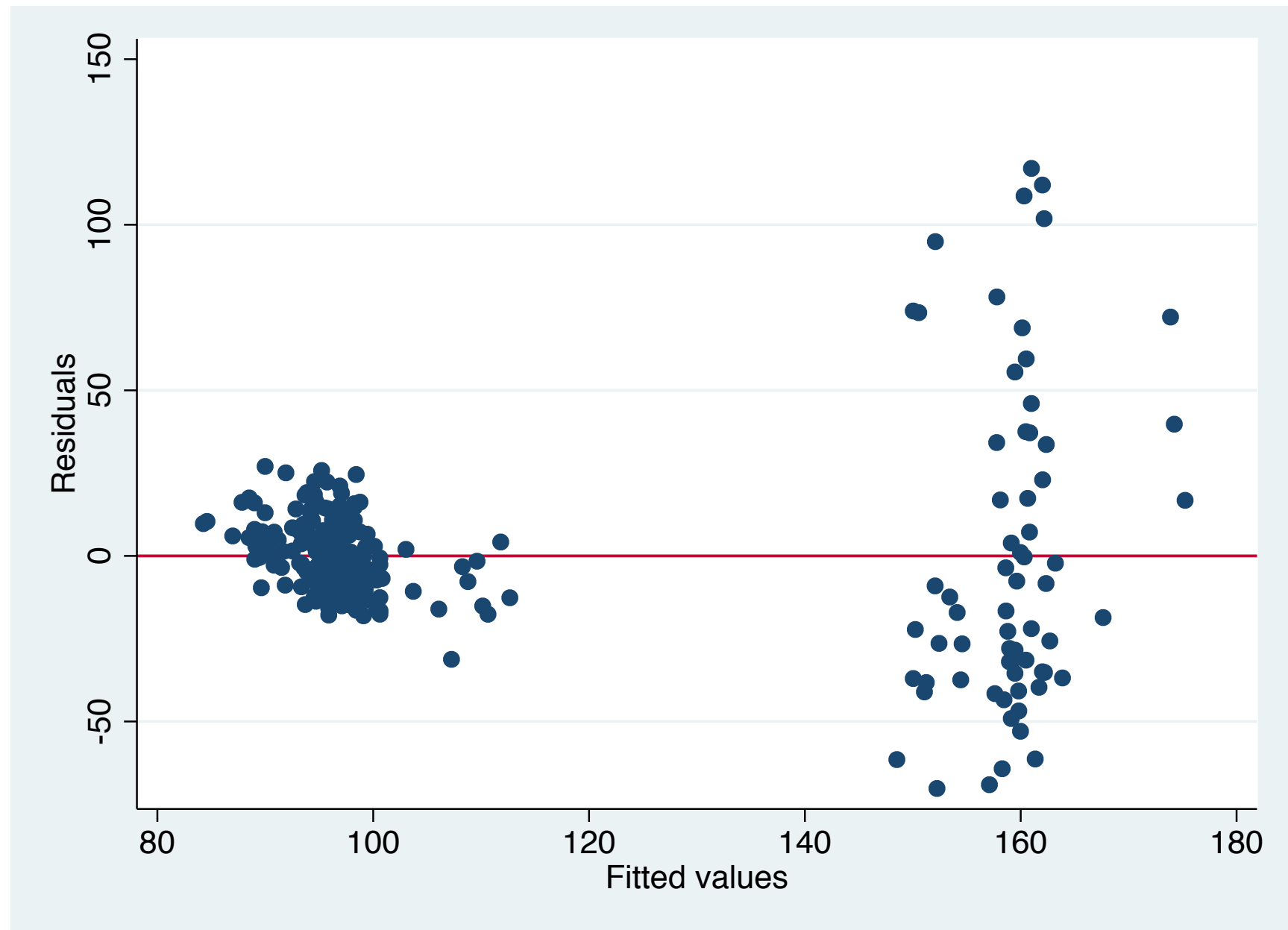
$\propto$ = “proportional to”

ρ = correlation coefficient

Example: model for mean glucose HERS

```
*  
. use hersdata, clear  
* take a 10% sample to make visualization easier (for illustration)  
. sample 10  
(2487 observations deleted)  
* untransformed outcome - fit model & plot residuals  
. quietly regress glucose diabetes i.physact age i.raceth smoking drinkany  
. rvfplot, yline(0)  
. graph export ./figures/rvp1.pdf, replace
```

RVF plot - untransformed outcome



Comparing residual variance by subgroup (un-transformed outcome)

```
. tabstat resid1, by(physact) stat(n var) nototal
```

Summary for variables: resid1

by categories of: physact (comparative physical activity)

physact	N	variance
much less active	26	1057.983
somewhat less ac	46	792.155
about as active	87	731.2216
somewhat more ac	85	554.0864
much more active	32	107.7953

```
. tabstat resid1, by(diabetes) stat(n var) nototal
```

Summary for variables: resid1

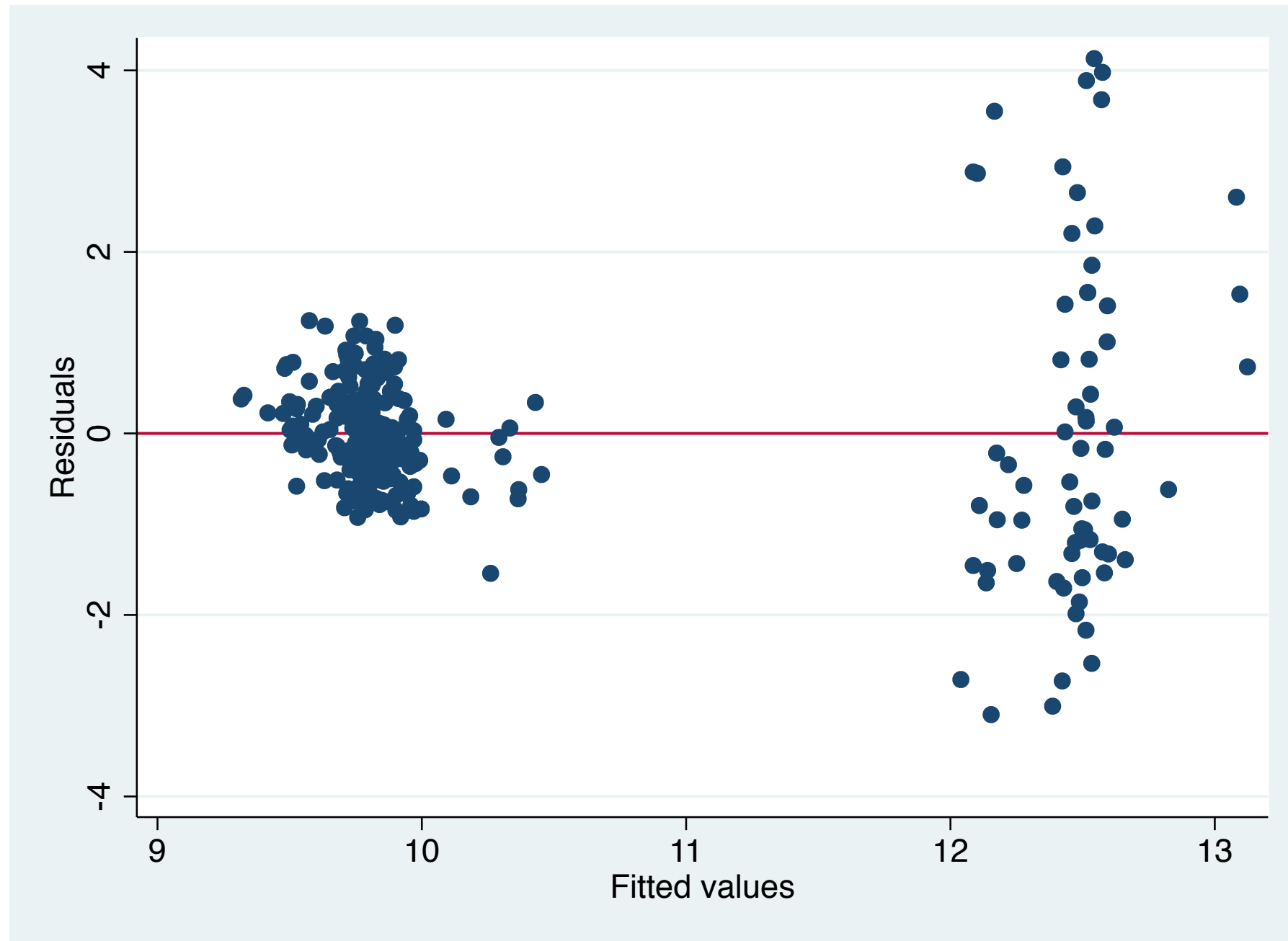
by categories of: diabetes (diabetes)

diabetes	N	variance
no	196	30.45264
yes	80	53.23272

Example: model for mean glucose HERS - square root transformed

```
* square root transformed outcome - fit model & plot residuals
. gen sqrt_glucose = sqrt(glucose)
. quietly regress sqrt_glucose diabetes i.physact age i.raceth smoking drinkany
. rvfplot, yline(0)
. graph export ./figures/rvp2.pdf, replace
```

RVF plot - square root transformed outcome



Solution: use robust SEs

```
. regress glucose diabetes i.physact age i.raceth smoking drinkany, vce(robust)
. . .
```

glucose		Coef.	Robust Std. Err.	t	P> t	[95% Conf. Interval]	
diabetes		55.32816	5.704065	9.70	0.000	44.09711	66.55922
physact							
somewhat less active		.5986391	7.670311	0.08	0.938	-14.50387	15.70115
about as active		6.51184	7.519767	0.87	0.387	-8.294252	21.31793
somewhat more active		2.873804	7.282648	0.39	0.693	-11.46541	17.21302
much more active		.4625191	6.907942	0.07	0.947	-13.13892	14.06396
age		-.3130465	.2466262	-1.27	0.205	-.7986428	.1725497
raceth							
African American		9.907849	7.805314	1.27	0.205	-5.460473	25.27617
Other		22.48085	15.08384	1.49	0.137	-7.218569	52.18027
smoking		-4.696382	4.223875	-1.11	0.267	-13.01301	3.620243
drinkany		6.649252	3.427625	1.94	0.053	-.0995925	13.3981
_cons		112.8064	16.89753	6.68	0.000	79.53592	146.0769

... or use more conservative robust SEs

```
. regress glucose diabetes i.physact age i.raceth smoking drinkany, vce(hc3)
. . .
```

glucose		Robust HC3		t	P> t	[95% Conf. Interval]	
		Coef.	Std. Err.				
	diabetes	55.32816	5.838609	9.48	0.000	43.8322	66.82413
	physact						
somewhat less active		.5986391	8.082405	0.07	0.941	-15.31526	16.51254
about as active		6.51184	7.877636	0.83	0.409	-8.998881	22.02256
somewhat more active		2.873804	7.619965	0.38	0.706	-12.12957	17.87718
much more active		.4625191	7.247594	0.06	0.949	-13.80768	14.73271
	age	-.3130465	.2557038	-1.22	0.222	-.8165162	.1904231
	raceth						
African American		9.907849	8.189902	1.21	0.227	-6.21771	26.03341
Other		22.48085	16.98321	1.32	0.187	-10.95835	55.92005
	smoking	-4.696382	4.444625	-1.06	0.292	-13.44765	4.054891
	drinkany	6.649252	3.505505	1.90	0.059	-.2529339	13.55144
	_cons	112.8064	17.51732	6.44	0.000	78.31558	147.2972

- *Note:* hc3 option provides more conservative robust SE's that may perform better for models with pronounced non-constant variance and/or smaller samples sizes (compare SEs and CIs with previous slide)

Solution: Generalized linear models (GLM)

<i>outcome type</i>	<i>distribution</i>	<i>variance-mean relationship</i>
continuous	normal	constant
continuous	gamma	proportional ($\sigma^2 \propto \mu$)
binary	binomial	$\sigma^2 = n \mu(1 - \mu)$
binary	clustered binomial	$\sigma^2 \propto n \mu(1 - \mu)$
count	Poisson	equal ($\sigma^2 = \mu$)
count	overdispersed Poisson	$\sigma^2 \propto \mu$
count	negative binomial	$\sigma^2 = \mu + \mu^2/k$

Note: GLMs will be covered in Biostat 209 (also, see table 8.8 in text)

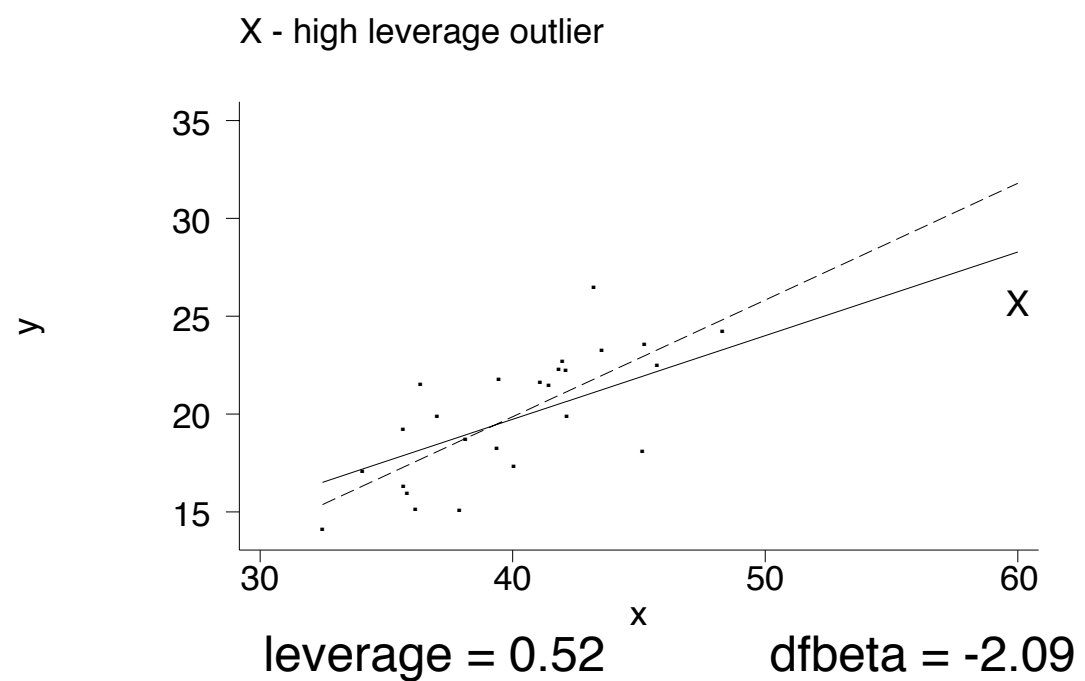
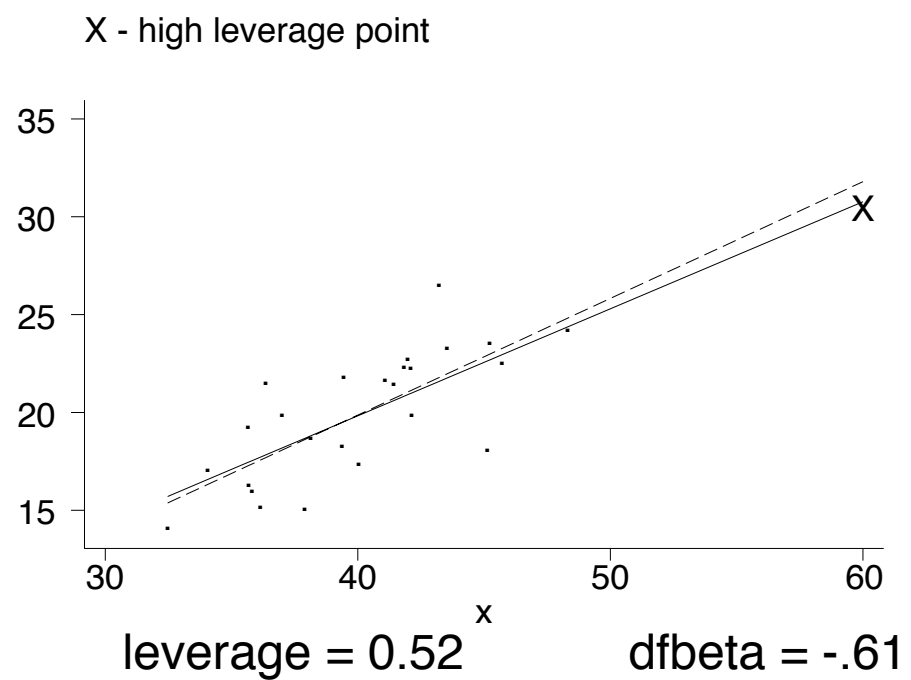
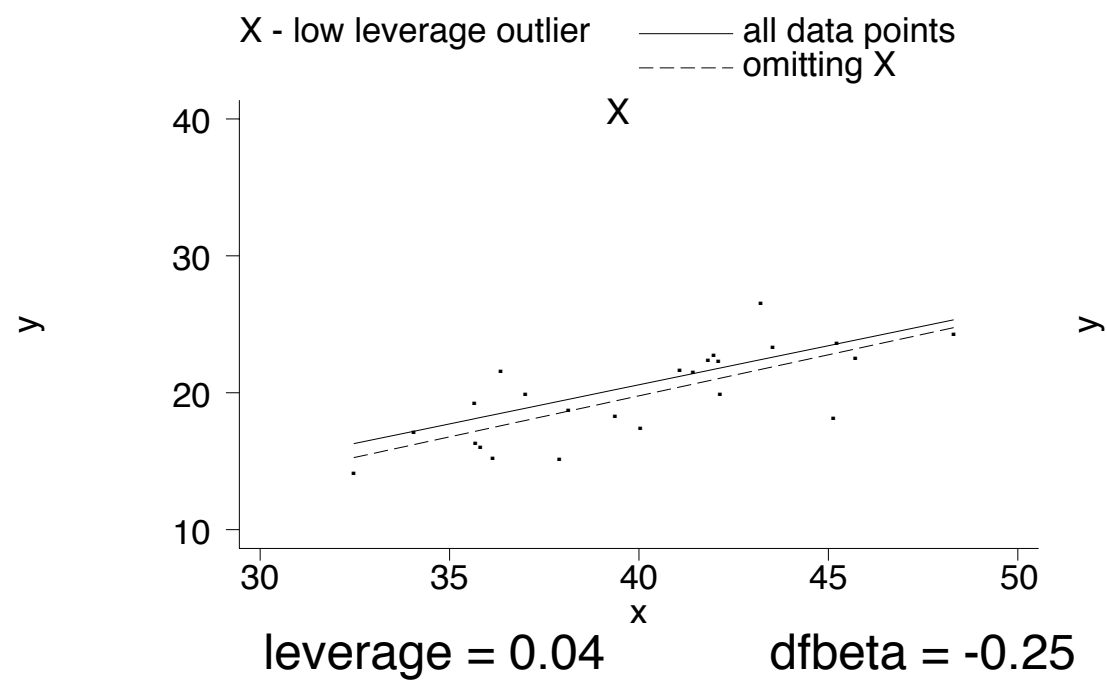
Checking constant variance: summary

- Diagnostics: funnel shapes in RVP plot, variable n , SD across subgroups
- Solutions: transform outcome, use robust SEs or GLMs

Checking the model: high leverage and influential points

- *High-leverage:*
 - ≥ 1 extreme predictor, or anomalous combination
 - potential to influence coefficient estimates unduly
- *Influential:*
 - high-leverage plus big impact on coefficients
- Inferences that are highly dependent on a few influential observations are potentially misleading

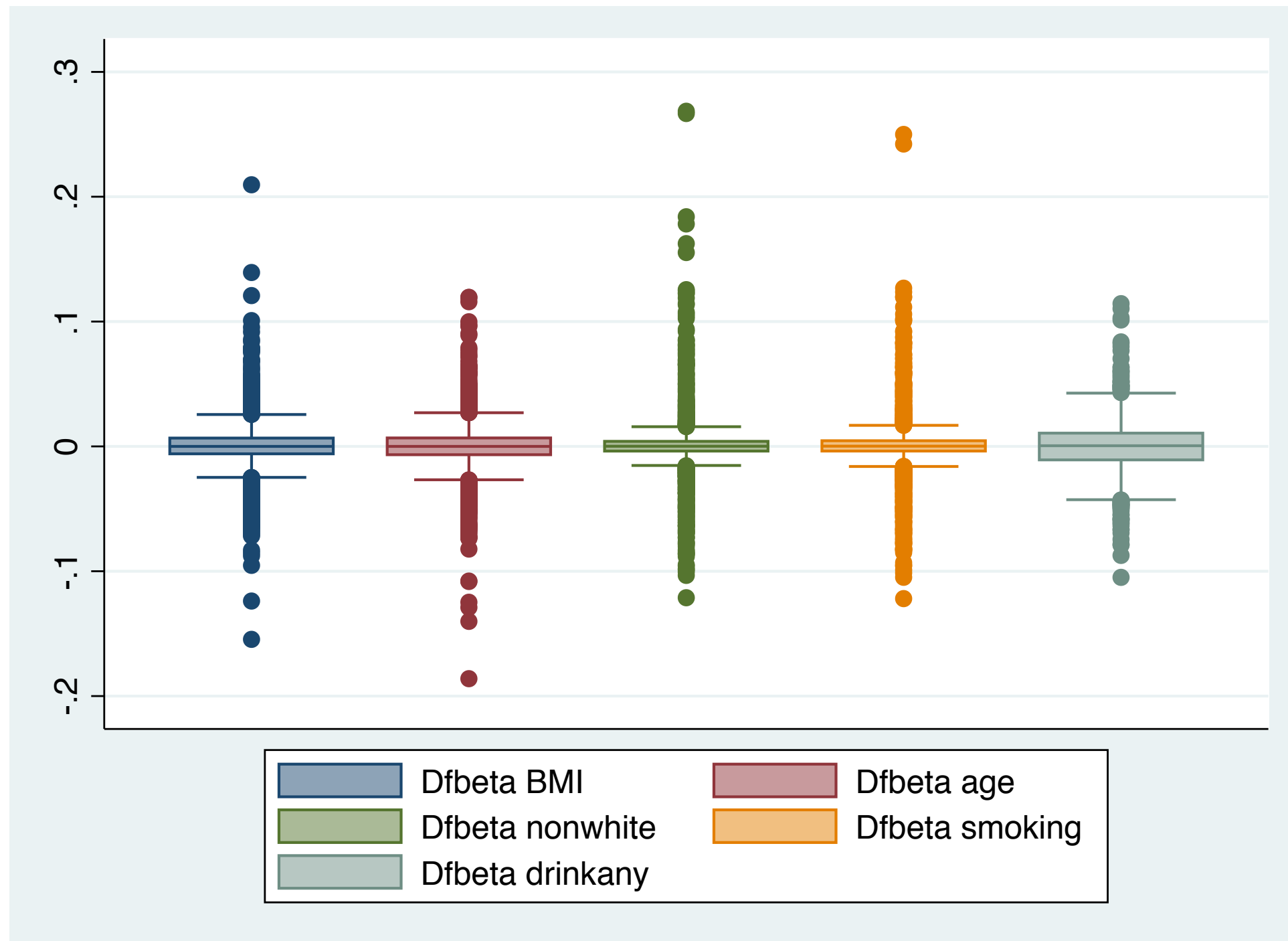
Examples: Simple outlier, high leverage, high influence



Diagnostics: boxplots of dfbeta statistics

- dfbeta statistics measure changes in each β_j when each data point is omitted
- Defined for each observation and predictor in model
- Check for outliers in boxplots of dfbetas

Boxplots of dfbetas for BMI - LDL model (slide 27)



Solution:

- Identify up to 10 observations with biggest dfbeta values

* Stata statements to list observations with large dfbeta (yields four observations)

* Statement for first predictor (BMI) below (repeat for other predictors)

```
. list id BMI age nonwhite smoking drinkany _dfbeta_1 if abs(_dfbeta_1) > 0.2 &  
~missing(_dfbeta_1)
```

	id	BMI	age	nonwhite	smoking	drinkany	_dfbeta_1
70.	70	28.74	63	yes	no	no	.2666293
1768.	1768	28.22	61	yes	yes	no	.2684654
1020.	1020	26.22	58	no	yes	yes	.2422048
1768.	1768	28.22	61	yes	yes	no	.2498547

- Check for data errors or other anomaly
- Re-fit model without influential points, reassess conclusions, report sensitivities
- Consider deleting influential points if they represent a different population

Sensitivity of LDL model to 4 influential points with $df\beta a > 0.2$ in absolute value

<i>predictor</i>	<i>all observations</i>		<i>omitting 4 points</i>	
	β	p	β	p
BMI	0.36	0.0007	0.34	0.010
Age	-1.89	0.090	-1.86	0.090
Nonwhite	5.22	0.025	4.19	0.066
Smoking	4.75	0.032	3.78	0.072
Alcohol use	-2.72	0.069	-2.64	0.072

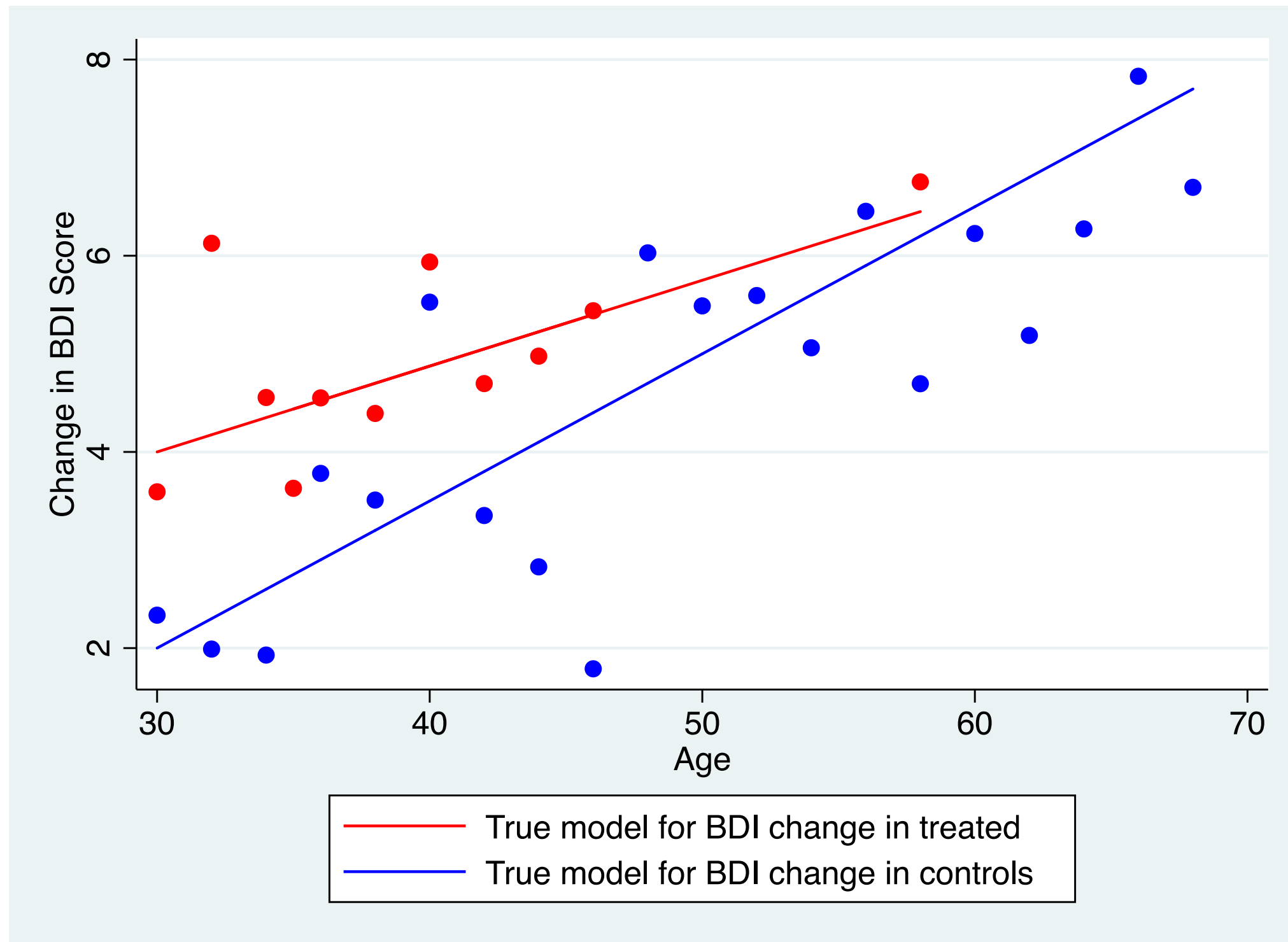
Checking influential points: summary

- Diagnostics: boxplots of $dfbetas$
- Solutions: fix errors, conduct sensitivity analyses omitting influential points

Checking the model: predictor/covariate overlap

- Observational analysis of binary exposure problematic if exposed, unexposed too unlike
- Lack of overlap makes true model hard to find, especially in small datasets
 - conclusions about effect of exposure may be based on *extrapolation*
- Comparing each covariate in exposed and unexposed may not be enough, because covariates are correlated:
 - some *combinations* of predictors may be unrepresented in one group

Example: Lack of age overlap in model for effect of treatment on Beck Depression Inventory score



No power to detect interaction

```
. regress del_bdi i.treatment#c.age
```

Source	SS	df	MS	Number of obs = 31			
Model	46.3692007	3	15.4564002	F(3, 27)	=	15.42	
Residual	27.0583639	27	1.00216163	Prob > F	=	0.0000	
Total	73.4275647	30	2.44758549	R-squared	=	0.6315	
				Adj R-squared	=	0.5906	
				Root MSE	=	1.0011	

del_bdi	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
1.treatment	3.217112	1.88746	1.70	0.100	-.6556366	7.08986
age	.1247361	.0194101	6.43	0.000	.0849098	.1645623
treatment#c.age						
1	-.0429515	.0445653	-0.96	0.344	-.1343918	.0484889
_cons	-1.483581	.9770828	-1.52	0.141	-3.488389	.5212275

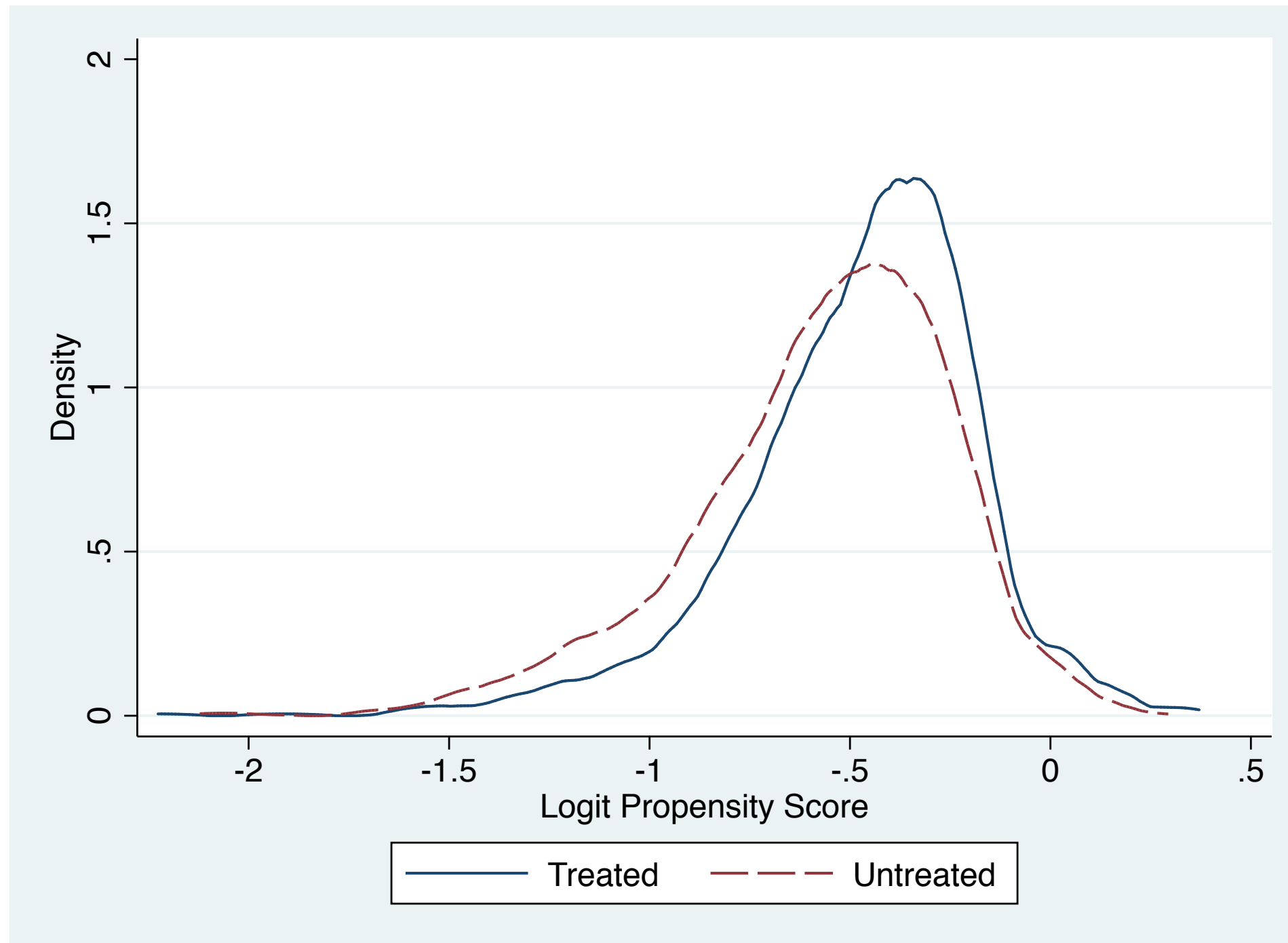
Diagnosing lack of overlap

- Compare mean, quartiles, range of covariates in exposed and unexposed
- Use *propensity scores*
 - fit logistic model for primary predictor
 - include an MSAS for the exposure-outcome relationship
 - capture non-linearities and interactions
 - get fitted values (on linear predictor or probability scale)
 - plot the results by primary predictor and check overlap

Propensity score model for statin use

```
. * logistic model for statin use
. quietly logistic statins agesp* i.raceth i.educ_cat i.smoking##i.lessactive diabetes
. * calculate logit propensity score
. predict logit_ps, xb
. * density plots of logit scores in statin users and non-users
. twoway (kdensity logit_ps if statins==1, area(1) lpattern(solid)) ///
>         (kdensity logit_ps if statins==0, area(1) lpattern(longdash)), ///
>         ytitle("Density") xtitle("Logit Propensity Score") ///
>         legend(order(1 "Treated" 2 "Untreated")) ///
>         saving(pscores, replace)
```

Overlap diagnostics for statin use in HERS

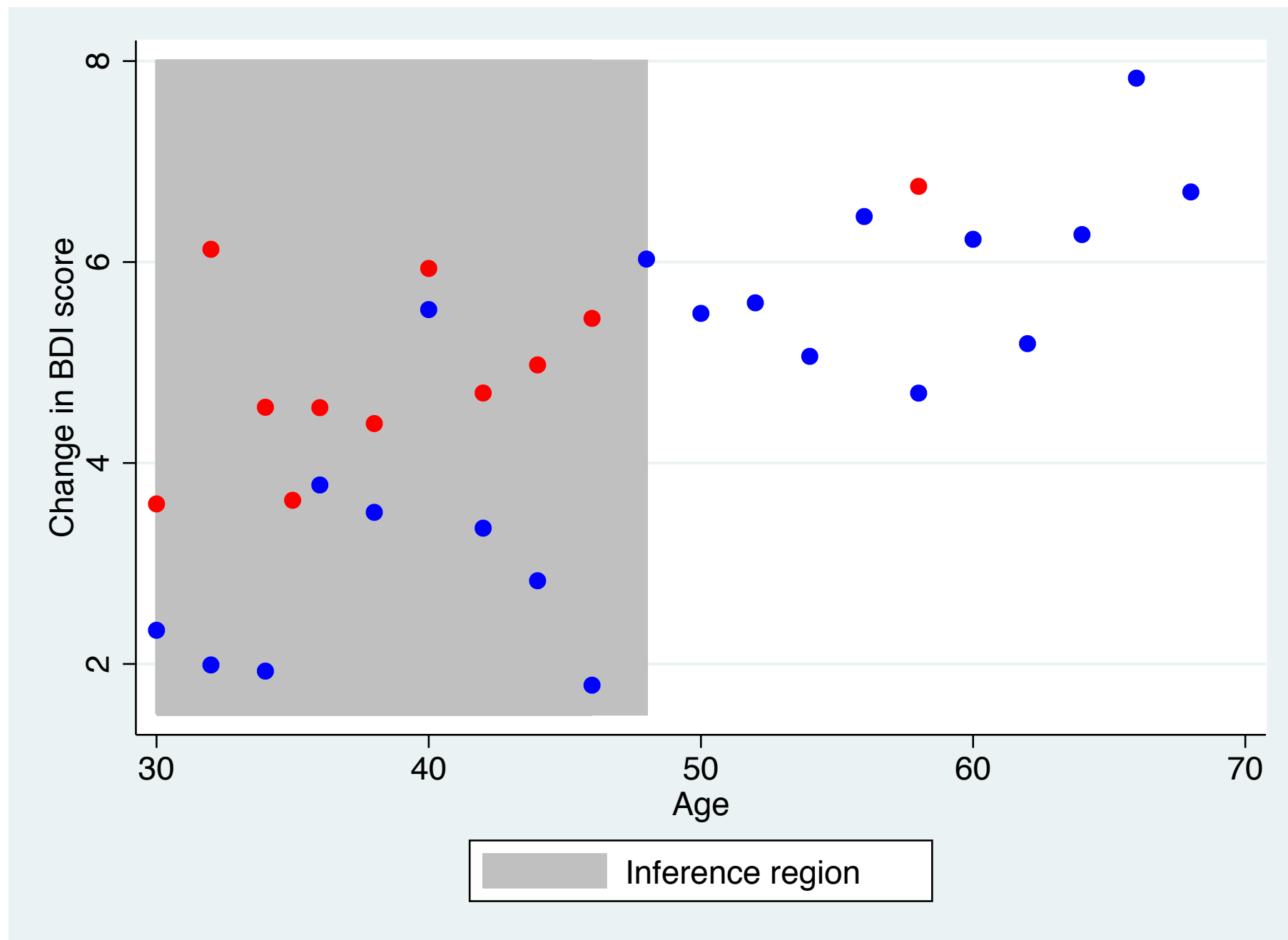


Overlap is generally quite good (thanks to treatment randomization)

Solution: lack of overlap

- Restrict inference to region of good overlap
- Match on prognostic covariates or propensity scores (covered in Biostat 215)

Restricting inference to region of overlap (BDI example)



Compare conclusions based on observations with sufficient overlap to results from full sample

Model checking: to transform or not

- Transformations can help meet assumptions
 - ─ but make results harder to interpret
- If violations mild, results robust, reasonable not to transform
- If conclusions change substantially after transformation
 - ─ model that meets assumptions better is more reliable

Model checking: summary

- Non-linearity:
 - Diagnostics: curved Lowess smooth in CPR or RVP plot
 - Solutions: transform predictor, including splines
- Non-normality:
 - Diagnostics: curvature in QQ-plot
 - Solutions: transform outcome, use bootstrap CIs, GLM or ordinal model

Model checking: summary

- Non-constant variance:
 - Diagnostics: funnel shapes in RVP plot, SDs differ across unequal size subgroups
 - Solutions: transform outcome, use GLM, robust SEs
- Influential points:
 - Diagnostics: boxplots of $df\beta$ statistics
- – Solutions: identify up to 10 influential points, correct data errors, omit influential points if justifiable, present sensitivity analysis