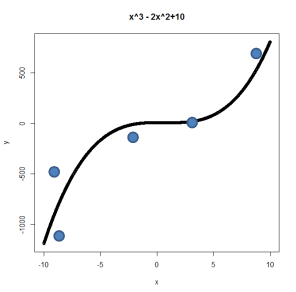
POLYNOMIAL REGRESSION



$DV = y$, $IV = x$ (one independent variable)

Input data : $\{(x_1, y_1), (x_2, y_2), ..., (x_k, y_k)\}$

Polynomial model is

$$y_i = \beta_0 + \beta_1 x + \beta_2 x^2 + ... + \beta_k x^k + e_i, \quad i = 1, 2, ..., n$$

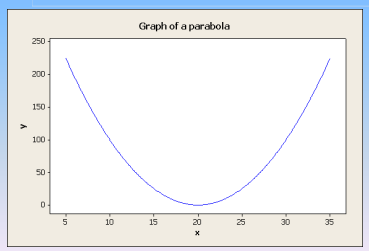
$k = 2$: quadtratic model

$k = 3$: cubic model

NOTE: Even though the model is NONLINEAR in x, it is still LINEAR in the unknown β's, and therefore still falls under MLR.

ASSUMPTIONS:

1) *Mean value of y*, given $x=x_i$ is given by the above polynomial.
2) The error terms e_i are independent with mean 0, common variance σ^2.

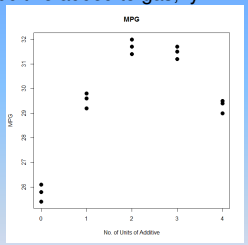


In a regression situation, the graph of data would depend upon the range of x-values chosen in the experiment.

Solved Example 4.11/page 168, Bowerman, O'Connell, Koehler, 4-th Edition – Forecasting, Time Series, and Regression :

x = number of units of additive added to gas, y = MPG

No_units	MPG
0	25.8
0	26.1
0	25.4
1	29.6
1	29.2
1	29.8
2	32
2	31.4
2	31.7
3	31.7
3	31.5
3	31.2
4	29.4
4	29
4	29.5



Scatter plot suggests a quadratic model (single bend in curve implies 2-nd degree polynomial).


```
# R-code for fitting a 2-nd degree equation
d <- read.csv("MPG.csv",header=TRUE)

# scatterplot of data
plot(d$No_units,d$MPG, xlab="No. of Units of
Additive", ylab="MPG",
main="MPG",cex=1.5,pch=16)

# fit a 2-nd degree polynomial
u <- d$No_units
u2 <- u**2
lm1 <- lm(d$MPG~u+u2)
summary(lm1)
```


In MINITAB, we can fit a straight line, quadratic, and cubic model in 2 ways:

1) Using the procedure REGRESSION/REGRESSION:
   (a) For fitting a quadratic, first create a column for x^2 using CALC in MINITAB.

Regression Analysis: MPG versus No_units, No_units_sqr

The regression equation is
MPG = 25.7 + 4.98 No_units - 1.02 No_units_sqr

	Estimate	Std.Error	t	P-value
(Intercept)	25.72	0.16	165.43	0.00
u	4.98	0.18	27.02	0.00
u2	-1.02	0.04	-23.09	0.00

Each P < .05
So each term
Is significant.

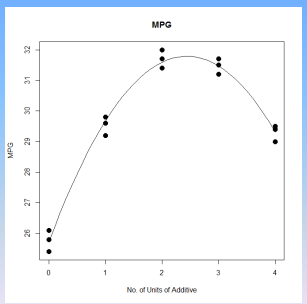
Residual standard error: 0.2861 on 12 degrees of freedom
Multiple R-squared: 0.9857, Adjusted R-squared: 0.9834
F-statistic: 414.9 on 2 and 12 DF, p-value: 8.389e-12

Each StdError is small relative to its estimate, so regression model looks good.

```
# R-code for scatterplot of data with fitted equation
# compute fitted values
beta0 <- lm1$coefficients[1]
beta1 <- lm1$coefficients[2]
beta2 <- lm1$coefficients[3]

# plot fitted values on the scatterplot
uu <- seq(0,4,by=.1)
yhat <- beta0 + beta1*uu + beta2*uu**2
lines(uu,yhat)
```


2) Using the procedure REGRESSION/FITTED LINE PLOT



EFFECT OF INTERACTION

Example:
For the given data, fit a MLR model.

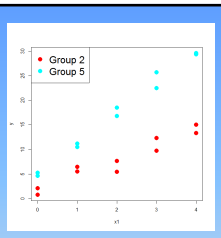
x1	x2	y
0	2	2.0693
0	2	0.7322
0	5	5.2437
0	5	4.6137
1	2	6.4264
1	2	5.4906
1	5	10.5035
1	5	11.1925
2	2	7.6372
2	2	5.4523
2	5	18.47
2	5	16.7838
3	2	12.318
3	2	9.7157
3	5	25.7062
3	5	22.4614
4	2	15.0363
4	2	13.348
4	5	29.65
4	5	29.3479


```
# R-code for scatterplot of y vs x1 by x2

d1 <- read.csv("interaction.csv",header=TRUE)

plot(y ~ x1, pch=16, col=x2, data = d1, cex=2)

legend('topleft',paste('Group',c(2,5),sep=" "),col=c(2,5),pch=16,cex=2)
```



Since the slopes of graphs of y vs x1 for x2=2 and x2=5 are quite different (i.e., the two graphs are not parallel), there is an interaction effect of x1 and x2 on y.

Regression Analysis : y versus x1, x2

	Estimate	Std.Error	t	P-value
(Intercept)	-7.8623	1.7539	-4.483	0.00
x1	4.6509	0.4268	10.897	0.00
x2	3.1916	0.4024	7.932	0.00

Multiple R-squared: 0.9144, Adjusted R-squared: 0.9044

Regression Analysis: y versus x1, x2, x1x2

	Estimate	Std.Error	t	P-value	value
(Intercept)	-0.4574	1.1826	-0.387	0.70	
x1	0.9484	0.4828	1.964	0.07.	
x2	1.0759	0.3106	3.464	0.00**	
x1x2	1.0578	0.1268	8.343	0.00***	

Multiple R-squared: 0.984, Adjusted R-squared: 0.981

Since x1 is not significant at 5% (see slide 11), we drop x1: in R
y ~ x2 + x1x2, data = d1)

	Estimate	Std.Error	t	P-value	value
(Intercept)	1.43949	0.73796	1.951	0.07.	
x2	0.61801	0.22184	2.786	0.01*	
x1x2	1.28677	0.05398	23.838	0.00***	
Multiple	R-squared:	0.9802,	Adjusted	R-squared:	0.9778

Running a no-intercept model, we get:

# no intercept model in R
lm0 <- lm(y~0+x1+x2+x1x2,data=d1)
summary(lm0)

	Estimate	Std.Error	t	P-value
x1	0.79597	0.27168	2.93	0.01
x2	0.96548	0.11924	8.097	0.00
x1x2	1.09464	0.08167	13.403	0.00

Do you notice something different in this case?
There is no R^2 – value for a model with no-intercept!

Compute correlatio coefficients between y and yhat from the four models:

> cor(d1$y,lm0$fitted.values)
[1] 0.9919091
> cor(d1y,lmafitted.values)
[1] 0.956258
> cor(d1y,lmbfitted.values)
[1] 0.9919717
> cor(d1y,lmcfitted.values)
[1] 0.9900255

Models 1, 3, and 4 are very close to each other and fit data very well - this can be seen from the 4 plots on the next slide.

#R-code
par(mfrow=c(2,2))

plot(d1$y,lm0$fitted.values,main="no intercept: y=b1x1+b2x2+b3x1x2")

plot(d1y,lmafitted.value,main="no interaction: y=b0+b1x1+b2x2")

plot(d1y,lmbfitted.values,main="y=b0+b2x2+b3x1x2")

plot(d1y,lmcfitted.values,main="y=b0+b1x1+b2x2+b3x1x2")