DOCUMENT(); # This should be the first executable line in the problem.

loadMacros(

 "PGstandard.pl",

 "MathObjects.pl",

 "answerHints.pl",

 "PGchoicemacros.pl",

 "PGcourse.pl",

 "parserMultiAnswer.pl",

 "parserPopUp.pl",

);

TEXT(beginproblem());

#######only linear and quadratic are calculated in problem

#################################################

# Set-up

$points = 5;

$x[0] = 10;

$x[1] = 20;

$x[2] = 30;

$x[3] = 40;

$x[4] = 45;

$y[0] = 16.0;

$y[1] = 20.3;

$y[2] = 21.9;

$y[3] = 23.3;

$y[4] = 23.9;

$year = 2020;

$mpg = 100;

#############linear regression

$sx =0;

$sy =0;

$sxy =0;

$sx2 =0;

$sy2 =0;

$sx3 =0;

$sx4 =0;

$sx2y =0;

$meanx =0;

$meany =0;

$sse =0;

$sst =0;

for($i=0; $i<$points; $i++) {

#####sum of x

 $sx = $sx + $x[$i];

#####sum of y

 $sy = $sy + $y[$i];

#####sum of xy

 $sxy = $sxy + ($x[$i]\*$y[$i]);

#####sum of x^2

 $sx2 = $sx2 + ($x[$i]\*$x[$i]);

#####sum of y^2

 $sy2 = $sy2 + ($y[$i]\*$y[$i]);

#####sum of x^3

 $sx3 = $sx3 + ($x[$i]\*$x[$i]\*$x[$i]);

#####sum of x^4

 $sx4 = $sx4 + ($x[$i]\*$x[$i]\*$x[$i]\*$x[$i]);

#####sum of x^2 y

 $sx2y = $sx2y + ($x[$i]\*$x[$i]\*$y[$i]);

#####for the mean

 $meanx = $meanx+$x[$i];

#####for the mean

 $meany = $meany+$y[$i];

}

######mean of x

$meanx = $meanx/$points;

######mean of y

$meany = $meany/$points;

######linear regression terms

$ssxy = $points\*$sxy-(($sx\*$sy));

$ssx = $points\*$sx2-($sx\*$sx);

$ssy = $points\*$sy2-($sy\*$sy);

########linear regression calculation

$r = sprintf("%0.5f",$ssxy/sqrt($ssx\*$ssy));

$r2 = sprintf("%0.3f",$r\*$r);

$b1 = sprintf("%0.4f",$ssxy/$ssx);

$b0 = sprintf("%0.4f",$meany-$b1\*$meanx);

######quadratic regression terms

$qa = (($points\*$sx2y- $sx2\*$sy)\*($points\*$sx2 -$sx\*$sx)-

 ($points\*$sxy-$sx\*$sy)\*($points\*$sx3 -$sx\*$sx2))/

 (($points\*$sx2 -$sx\*$sx)\*($points\*$sx4 -($sx2\*$sx2))-

 ($points\*$sx3-$sx\*$sx2)\*($points\*$sx3-$sx\*$sx2));

$a = sprintf("%0.4f",$qa);

$qb = (-($points\*$sx2y- $sx2\*$sy)\*($points\*$sx3-$sx\*$sx2)+

 ($points\*$sxy-$sx\*$sy)\*($points\*$sx4 -($sx2\*$sx2)))/

 (($points\*$sx2 -$sx\*$sx)\*($points\*$sx4 -($sx2\*$sx2))-

 ($points\*$sx3-$sx\*$sx2)\*($points\*$sx3-$sx\*$sx2));

$b = sprintf("%0.4f",$qb);

$qc = ($sy/$points)-$qb\*($sx/$points)-$qa\*($sx2/$points);

$c = sprintf("%0.4f",$qc);

######for quadratic r^2

for($i=0; $i<$points; $i++) {

######for SST

 $sst = $sst + ($y[$i]-$meany)\*($y[$i]-$meany);

#####for SSE

 $sse = $sse + ($y[$i]-$qa\*$x[$i]\*$x[$i] -$qb\*$x[$i] -$qc)\*($y[$i]-$qa\*$x[$i]\*$x[$i] -$qb\*$x[$i] -$qc);

}

$qr = sprintf("%0.3f",1-($sse/$sst));

#################################################

####multianswer

Context()->strings->add("no solution"=>{},

 "no solutions"=>{alias=>'no solution'},

 "none"=>{alias=>'no solution'});

Context()->strings->add("linear"=>{});

Context()->strings->add("exponential"=>{});

Context()->strings->add("logarithmic"=>{});

Context()->strings->add("logistic"=>{});

Context()->strings->add("quadratic"=>{});

$ans\_5 = PopUp(["select","linear","quadratic","exponential","logistic","logarithmic"], "logarithmic");

$ans\_6 = $b0 + $b1\*($year-1970);

$ans\_7 = ($mpg-$b0)/($b1);

$multians = MultiAnswer($ans\_5, $ans\_6, $ans\_7)->with(

 singleResult => 0, allowBlankAnswers => 1,

 checker => sub {

 my ( $correct, $student, $self ) = @\_;

 my ( $a1s, $a2s, $a3s ) = @{$student};

 my ( $a1, $a2, $a3 ) = @{$correct};

 if ( $a1s->value == "linear"){

 if ($a2s == $b0+$b1\*($year-1970) && $a3s == ($mpg-$b0)/$b1) {$self->setMessage(1,"A line is not the best choice.");

 return [1,1,1]}

 elsif ($a2s == $b0+$b1\*($year-1970) && $a3s != ($mpg-$b0)/$b1) {$self->setMessage(1,"A line is not the best choice.");

 $self->setMessage(3,"Check your calculation.");

 return [1,1,0]}

 elsif ($a2s != $b0+$b1\*($year-1970) && $a3s == ($mpg-$b0)/$b1) {$self->setMessage(1,"A line is not the best choice.");

 $self->setMessage(2,"Check your calculation.");

 return [1,0,1]}

 else {$self->setMessage(1,"A line is not the best choice.");

 $self->setMessage(2,"Check your calculation.");

 $self->setMessage(3,"Check your calculation.");

 return [1,0,0]}}

 elsif ( $a1s->value == "quadratic"){

 if ($a2s == $a\*($year-1970)\*\*2 +$b\*($year-1970) +$c && $a3s == "no solution") {

 return [1,1,1]}

 elsif ($a2s == $a\*($year-1970)\*\*2 +$b\*($year-1970) +$c && $a3s != "no solution") {$self->setMessage(3,"Check your calculation.");

 return [1,1,0]}

 elsif ($a2s != $a\*($year-1970)\*\*2 +$b\*($year-1970) +$c && $a3s == "no solution") {$self->setMessage(2,"Check your calculation.");

 return [1,0,1]}

 else {$self->setMessage(2,"Check your calculation.");

 $self->setMessage(3,"Check your calculation.");

 return [1,0,0]}}

 elsif ( $a1s->value == "exponential"){

 if ($a2s == 15.3285\*(1.0107\*\*($year-1970)) && $a3s == ln($mpg/15.3285)/ln(1.0107)) {$self->setMessage(1,"A exponential is not a good choice.");

 return [0,1,1]}

 elsif ($a2s == 15.3285\*(1.0107\*\*($year-1970)) && $a3s != ln($mpg/15.3285)/ln(1.0107)) {$self->setMessage(1,"A exponental is not a good choice.");

$self->setMessage(3,"Check your calculation.");

 return [0,1,0]}

 elsif ($a2s != 15.3285\*(1.0107\*\*($year-1970)) && $a3s == ln($mpg/15.3285)/ln(1.0107)) {$self->setMessage(1,"A exponential is not a good choice.");

$self->setMessage(2,"Check your calculation.");

 return [0,0,1]}

 else {$self->setMessage(1,"A exponential is not a good choice.");

 $self->setMessage(2,"Check your calculation.");

 $self->setMessage(3,"Check your calculation.");

 return [0,0,0]}}

 elsif ( $a1s->value == "logistic"){

 if ($a2s == 24.2704/(1+1.2059\*exp(-0.08638\*($year-1970))) && $a3s == "no solution") {

 return [1,1,1]}

 elsif ($a2s == 24.2704/(1+1.2059\*exp(-0.08638\*($year-1970))) && $a3s != "no solution") {$self->setMessage(3,"Check your calculation.");

 return [1,1,0]}

 elsif ($a2s != 24.2704/(1+1.2059\*exp(-0.08638\*($year-1970))) && $a3s == "no solution") {$self->setMessage(2,"Check your calculation.");

 return [1,0,1]}

 else {$self->setMessage(2,"Check your calculation.");

 $self->setMessage(3,"Check your calculation.");

 return [1,0,0]}}

 elsif ( $a1s->value == "logarithmic"){

 if ($a2s == 4.3528+5.1643\*ln(($year-1970)) && $a3s == exp(($mpg-4.3528)/5.1643)) {

 return [1,1,1]}

 elsif ($a2s == 4.3528+5.1643\*ln(($year-1970)) && $a3s != exp(($mpg-4.3528)/5.1643)) {$self->setMessage(3,"Check your calculation.");

 return [1,1,0]}

 elsif ($a2s != 4.3528+5.1643\*ln(($year-1970)) && $a3s == exp(($mpg-4.3528)/5.1643)) {$self->setMessage(2,"Check your calculation.");

 return [1,0,1]}

 else {$self->setMessage(2,"Check your calculation.");

 $self->setMessage(3,"Check your calculation.");

 return [1,0,0]}}

 else {

 return [0,0,0];

 }

 }

);

# Main

BEGIN\_TEXT

$BBOLD Note: The problem covers problems 1-5 from Section 5.5 of College Algebra an Applied Approach. $EBOLD

$PAR

$BBOLD Note on Rounding: Your answers are dependent on rounding. Be sure to round as the problem specifies. $EBOLD

$PAR

$BR

The following questions use the following information from the U.S. Department

of Transportation.

$BR

\{begintable(2)\}

\{row("Years after 1970", "Avg. Number of MPG for U.S. Automobiles (light duty)")\}

\{row("\($x[0]\)", "\($y[0]\)")\}

\{row("\($x[1]\)", "\($y[1]\)")\}

\{row("\($x[2]\)", "\($y[2]\)")\}

\{row("\($x[3]\)", "\($y[3]\)")\}

\{row("\($x[4]\)", "\($y[4]\)")\}

\{endtable()\}

 $BR

(a) Find the linear , quadratic, exponential, logistic, and logarithmic regression models that gives the miles per gallon

as a function of years after 1970. (Round answers to four decimal places for the regression and 3 decimal places for the \(r^2\).)

$PAR

Linear: \(y = \) \{ans\_rule(15)\}

 with \(r^2 = \) \{ans\_rule(15)\}.

$PAR

$BR

Quadratic: \(y = \) \{ans\_rule(15)\}

 with \(r^2 = \) \{ans\_rule(15)\}.

$PAR

$BR

Exponential: \(y = \) \{ans\_rule(15)\}

 with \(r^2 = \) \{ans\_rule(15)\}.

$PAR

$BR

Logistic: \(y = \) \{ans\_rule(15)\}.

$PAR

$BR

Logarithmic: \(y = \) \{ans\_rule(15)\}

 with \(r^2 = \) \{ans\_rule(15)\}.

$PAR

$BR

$BR

b) Which of the better model to use?

$Par

$BR

The best model to use is \{$multians->menu()\}

$PAR

$BR

$BR

c) For the model chosen in the last part, what is the average miles per gallon expected to be in $year? (Round to 1 decimal place.)

$BR

The average miles per gallon in $year is predicted to be \{$multians->ans\_rule(15)\} mpg.

$PAR

$BR

$BR

d) For the model chosen, when will the miles per gallon be $mpg? (Depending on your, choice you may

have no solution or very large numbers. Round any decimals to one place.)

$BR

We predict that the mpg will be $mpg \{$multians->ans\_rule(15)\} years after 1970.

END\_TEXT

#################################################

# Answers

$showPartialCorrectAnswers = 1;

$ans\_1 = "$b0+$b1\*x";

ANS(fun\_cmp($ans\_1,tolType => 'absolute',

 tolerance => .002));

$ans\_2 = Compute($r2)->with(tolType=>'absolute', tolerance=>'0.002');

 ANS($ans\_2->cmp->withPostFilter(AnswerHints(

 sub {

 my ($correct,$student,$anshash) = @\_;

 return abs($student-$correct) < .01;

 } => ["Your answer is close. Try the calculation using more accuracy."])));

$ans\_3 = "$c + $b\*x +$a\*x^2";

ANS(fun\_cmp($ans\_3,tolType => 'absolute',

 tolerance => .0002));

$ans\_4 = Compute($qr)->with(tolType=>'absolute', tolerance=>'0.002');

 ANS($ans\_4->cmp->withPostFilter(AnswerHints(

 sub {

 my ($correct,$student,$anshash) = @\_;

 return abs($student-$correct) < .01;

 } => ["Your answer is close. Try the calculation using more accuracy."])));

$ans\_exp = "15.3285\*(1.0107)^x";

$ans\_expr2 = Compute(0.891)->with(tolType=>'absolute', tolerance=>'0.002');

ANS(fun\_cmp($ans\_exp,tolType => 'absolute',

 tolerance => .0002));

 ANS($ans\_expr2->cmp->withPostFilter(AnswerHints(

 sub {

 my ($correct,$student,$anshash) = @\_;

 return abs($student-$correct) < .01;

 } => ["Your answer is close. Try the calculation using more accuracy."])));

$ans\_logis = "24.2704/(1 + 1.2059\*e^(-0.08638\*x))";

ANS(fun\_cmp($ans\_logis,tolType => 'absolute',

 tolerance => .0002));

$ans\_ln = "4.3528+5.1643\*ln(x)";

$ans\_lnr2 = Compute(0.992)->with(tolType=>'absolute', tolerance=>'0.002');

ANS(fun\_cmp($ans\_ln,tolType => 'absolute',

 tolerance => .0002));

 ANS($ans\_lnr2->cmp->withPostFilter(AnswerHints(

 sub {

 my ($correct,$student,$anshash) = @\_;

 return abs($student-$correct) < .01;

 } => ["Your answer is close. Try the calculation using more accuracy."])));

ANS($multians->cmp());

ENDDOCUMENT(); # This should be the last executable line in the problem.