Precalc AI Lab

Modeling Football Kickoffs Using Quadratic Functions

Image from Pixabay, June 27, 2016. Source: `https://pixabay.com/photos/football-american-football-1481798/`

In this Lab you will write Python code and run it. You can write the code yourself or get an LLM to do it. Run the code in Google Colab. If it does not run correctly, rewrite until it works correctly.

Purpose of Lab

- Recognize different models for quadratic functions and realize what they are useful for.
- From a graph or equation of a quadratic function, find intercepts and vertex.
- Interpret zeros and vertex in context.
- Use features of a graph to write quadratic functions in different forms.

Context for Lab

In NFL or College football, a kickoff is a play that starts or restarts the game. The ball is placed on a tee at the kicking team's 35-yard line, and a player kicks it toward the opposing team, who attempts to catch and return the ball to gain field position. There are several strategies for kickoffs, all of which can be modeled using quadratic functions.

Part 1: Creating a Graph and Finding Equations to Model a Kickoff

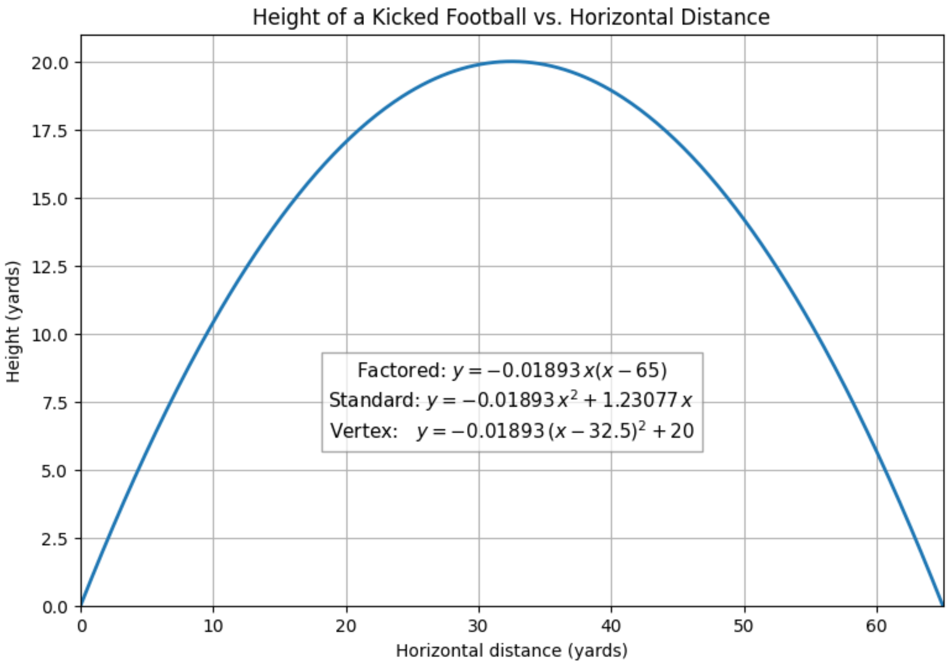
The most common kickoff strategy is to kick the ball as far down the field as possible without crossing the goal line. It also needs to be kicked high enough that it cannot be easily intercepted by players in between.

Use an LLM

Generate Python code to graph and find equations for a quadratic function that models the height of a ball after a kickoff in terms of its horizontal distance. We ask the LLM to:

- Generate Python code to create a quadratic model for the height of a ball after kickoff in a football game in terms of horizontal distance traveled.
- Ensure that the Python code is simple, easy to follow for a Precalculus student, and runs in Colab.
- Assume the ball travels 65 yards horizontally
- Assume the ball reaches a maximum height of 20 yards
- Create a graph that shows the quadratic function, but cut the graph off where the ball hits the ground
- Give the equation for the quadratic function in standard, vertex, and factored form underneath the graph.

We get the following graph and equations. The code is given at the end of the lab.



Part 2: Using and Interpreting Different Models

There are three commonly used forms for a quadratic function. Each of the different forms reveals certain information about a quadratic. We have

- **Standard form:** $f(x) = ax^2 + bx + c$
  - Can easily find the y-intercept $(0, c)$
- **Vertex form:** $f(x) = a(x-h)^2 + k$
  - Can easily find the vertex (h, k)
  - Can easily find the axis of symmetry $x = h$
- **Factored form:** $f(x) = a(x-r)(x-s)$
  - Can easily find the x-intercepts $(r, 0), (s, 0)$

Answer the following:

1. What is the y-intercept of the quadratic function found in Part 1? What does it tell you about the kickoff?
2. What is the vertex of the quadratic function found in Part 1? What does it tell you about the kickoff?
3. What are the x-intercepts of the quadratic function found in Part 1? What does it tell you about the kickoff?

Part 3: Creating a Graph and Finding Equations to Model a Kickoff Using a Different Strategy

Other kickoff strategies include:

- **Onside kick:** short, low kickoff where the kicking team attempts to recover the ball themselves. Usually reaches a height of about 2 yards and travels up to 12 yards.
- **Sky kick (or pooch kick):** high, short kickoff intended to give the kicking team time to get downfield. Usually reaches a height of about 18 yards and travels about 30 yards.
- **Squib kick:** the ball is kicked low and bounces along the ground (or stays low in the air) for about 30 yards reaching a max height of about 2 yards. It's meant to limit the chance of a long return by kicking away from

the deep returners.

Answer the following:

1. Between the four strategies (regular, onside, sky and squib), which do you expect to have a vertex at the highest coordinate? Why?
2. Between the four strategies, which do you expect to have the shortest distance between the x-intercepts? Why?

Use an LLM

Create python code that runs in Colab and gives an interactive graph of a quadratic modeling a kickoff in football. Have sliders for the maximum height and the length of the kick, and have it give the equation, in standard form, vertex form, and factored, at the bottom of the graph.

Answer the following with your interactive graph:

3. Choose one of the alternate kickoff strategies and use the interactive graph to sketch the graph of this kickoff, and to find the equation in standard form.
4. Notice that the number a in standard form is the same in both vertex form and factored form. This means you can use features of the graph to build the other forms. Specifically:
   a) What are the coordinates of the vertex? Use this to write down the vertex form for this quadratic.
   b) What are the x-intercepts? Use this to write down the factored form of the quadratic.

Further discussion

1. Make the math visible

Ask the LLM to walk you in all detail through the math it did to find the formulas used in the code. Make sure you tell it you are a Precalculus student and want to learn, so that it does not use any fancy math. Then, ask it to show you where that math is used in the code.

Provide an explanation in your own words of how the python code figures out the numbers for each quadratic form to display them.

2. Apply the math

Using what you learned in this lab, and just working with the formulas (without using your interactive graph), answer the following questions:

1. All three equations below describe the same kickoff path of a football. But they are written in different forms.

$$\text{Factored form: } f(x) = -0.03556x(x-45)$$

$$\text{Standard form: } f(x) = -0.03556x^2 + 1.6002x$$

$$\text{Vertex form: } f(x) = -0.03556(x-22.5)^2 + 18$$

   Answer the kickoff questions using whichever quadratic form makes it easiest.
   a) When does the ball leave the ground? When does it hit the ground again?
   b) How high does the ball go? How far does it travel horizontally?
   c) How high is the ball when it has traveled 15 yards?
2. A football is kicked from the ground at $(0,0)$. It reaches a maximum height of 25 and travels 55 yards. What quadratic function that models this? Show your steps and explain.
   a) give the factored form. What are the values of a, r and s?
   b) give the vertex form. What are the values of a, h, and k?

3. Strategic thinking

1. In the kickoff problem, why was the factored form the most useful? What important numbers (and what parts of the story) does it show right away?
2. You are modeling the path of a rocket that goes straight up and then back down. You do not care when it hits the ground, but you do care about the maximum height. Which form of a quadratic would you want to start with? Why?
3. If you are given an equation in standard form, and your only job is to find the output at x=50, would you bother rewriting it? Or is that form already the most convenient? Why?
4. If you were solving a real-world problem, how would you decide which form of the quadratic to start with? What clues in the problem would tell you whether factored, vertex, or standard form will make life easiest?

Sample Python Code

```python
import numpy as np
import matplotlib.pyplot as plt


# Parameters
R = 65.0 # total horizontal distance (yards)
h_max = 20.0 # maximum height (yards)


# Solve for 'a' in the factored form y = a * x * (x - R)
a = -4 * h_max / (R ** 2)
b = -a * R


# Create domain for trajectory
x = np.linspace(0, R, 400)
y = a * x * (x - R)


# Equations in LaTeX form (decimal coefficients)
factored_form_latex = r"$y = -0.01893\,x(x - 65)$"
standard_form_latex = r"$y = -0.01893\,x^2 + 1.23077\,x$"
vertex_form_latex = r"$y = -0.01893\,(x - 32.5)^2 + 20$"


# Plot
plt.figure(figsize=(9, 6))
plt.plot(x, y, linewidth=2)
plt.title("Height of a Kicked Football vs. Horizontal Distance")
plt.xlabel("Horizontal distance (yards)")
plt.ylabel("Height (yards)")
plt.grid(True)
plt.ylim(bottom=0)
plt.xlim(0, R)


# Place equations inside the plot near (32.5, 7.5)
equations_text = (f"Factored: {factored_form_latex}\n"
                  f"Standard: {standard_form_latex}\n"
                  f"Vertex: {vertex_form_latex}")
plt.text(32.5, 7.5, equations_text, fontsize=11, va="center", ha="center",
         bbox=dict(facecolor="white", alpha=0.7, edgecolor="gray"))


plt.show()
```