BEST ELLIPSES FITTING THE ORBITS OF PLANETS

CARMAN CATER

Abstract. Utilizing NASA HORIZONS Web-Interface we obtain the position vectors for some of the planets in our solar system and compute the parametrizations of their orbits around the sun. For sake of completeness, leading up to the results we discuss ways of finding best fitting lines, hyperplanes, and then the ellipse. For each we consider ways of measuring the error of the best-fit solution. There are many examples done in Mathematica illustrating the procedures and techniques discussed. If you are interested in any of the source code used for the procedures in Mathematica, please feel free to reach out to me at Email: carmancater@yahoo.com

1. Acknowledgment

I would like to thank Professor Oscar Perdomo of Central Connecticut State University for his patience in guiding me through this paper and providing most, if not all of the inspiration for the content of this paper. Without him this would not have been possible. The learning experience of writing this paper has been invaluable for me, and for that I owe him my sincerest gratitude.

2. Introduction

We begin our exploration of best fitting curves in Section 3 where we explore how to find the best line through a set of points in $\mathbb{R}^2$ with regards to minimizing the orthogonal distances from the points to the line. In this procedure we find an interesting result where we end up with two lines. The best fit line, as well as it's perpendicular. This can be used to compute some parameters of a planet's orbit when we notice that the two lines produced seem to be good approximations of the major and minor axes. This is left for future study. Section 4 provides an example of our best fitting line procedure.

Section 5 provides a technique when given a set of points in $\mathbb{R}^n$ to find the best hyperplane through the set of points that minimizes orthogonal distance. Interestingly, we find that this procedure gives perpendicular solutions similar to the perpendicular lines in the best line procedure. In the case of $\mathbb{R}^3$ we end up with three mutually orthogonal planes that appear to evenly distribute the set of points into octants. In particular, this procedure for finding the best plane through a set of points in $\mathbb{R}^3$ will be used when getting the parametrizations for the orbits of the planets. Section 6 provides an example of the best hyperplane procedure.

Date: September 22, 2021.

Section 7 covers three possible procedures for finding a best fitting ellipse to a set of points in $\mathbb{R}^2$. In Approach 1 we consider the equation of an ellipse of the form $\frac{(x-h)^2}{a^2} + \frac{(y-k)^2}{b^2} = 1$ and use minimization functions in Mathematica such as gradient descent to produce a best fit ellipse. We discuss some flaws of this approach. Approach 2 starts from the equation of a generic conic $Ax^2 + Bxy + Cy^2 + Dx + Ey + F = 0$, and after defining a way to measure the error from a point to the conic we find an exact solution set for the set of coefficients A, B, C, D, E, F using partial derivatives and matrix algebra. Approach 3 starts from the equation $(1 - e^2)Bx^2 + By^2 + Cx + Dy - 1 = 0$ thus forcing the eccentricity beforehand. The procedure of solving for the coefficients is identical to approach 2. Examples are provided following approaches 2 and 3. Also considered is one approach for computing the true error of the best fitting ellipse, using perpendicular distance from a point to the ellipse as our measure. The author finds this particularly interesting. For a large set of points the approach used is computationally intensive, but can easily be carried out in Mathematica.

In Section 8 we go into depth on the full procedure of finding the parametrizations of some of the planets in our solar system. Items covered in this section include

- Downloading and importing position vectors from the NASA HORIZONS Web-Interface
- Computing the plane of best fit
- Projecting the points onto the plane of best fit
- Computing the ellipse of best fit in $\mathbb{R}^3$
- Full example of the procedure

Lastly, Section 9 shows images of the orbits for Mercury, Venus, Earth, Mars, and Jupiter, as well as the full parametrizaitons in kilometers, and orbital parameters including the lengths of the semi-major/minor axes and eccentricity.

3. Best Line

Suppose we have a collection of points $S = \{p_1, \ldots, p_m\}$ in $\mathbb{R}^2$. We are interested in finding a line $\ell = \{(x, y) \in \mathbb{R}^2 : ax + by + c = 0\}$ that best fits these points. A natural way to find this line is to minimize the sum of orthogonal distances from each point p_i to the line ℓ. The orthogonal distance from p_i to ℓ is given by $d(p_i, \ell) = \dfrac{|ax_i + by_i + c|}{\sqrt{a^2 + b^2}}$.

If instead of (a, b) we use a unit vector $(\cos(\theta), \sin(\theta))$ our distance formula reduces to $d(p_i, \ell) = |x_i \cos(\theta) + y_i \sin(\theta) + c|$.

Therefore the function we are trying to minimize is $\sum_{i=1}^{m} |x_i \cos(\theta) + y_i \sin(\theta) + c|$. For convenience we will be minimizing the sum of the squared perpendicular distances

$$f(\theta, c) = \sum_{i=1}^{m} (x_i \cos(\theta) + y_i \sin(\theta) + c)^2$$

Experimentally this approach appears to provide a good approximation to $argmin\left(\sum_{i=1}^{m} |x_i \cos(\theta) + y_i \sin(\theta) + c|\right)$.

For a given point (x_i, y_i), a θ and a c the squared orthogonal distance from the point to the line is $(x_i \cos(\theta) + y_i \sin(\theta) + c)^2$ which is what we will refer to as the error. A direct computation shows that if $p_i = (x_i, y_i)$ and we define

$$X = (x_1, \dots, x_m) \in \mathbb{R}^m, \quad Y = (y_1, \dots, y_m) \in \mathbb{R}^m \quad u = (1, \dots, 1) \in \mathbb{R}^m$$

then we can re-write f as follows

$$f(\theta, c) = |X|^2 \cos^2(\theta) + |Y|^2 \sin^2(\theta) + (X \cdot Y) \sin(2\theta) \\ + (X \cdot u) 2c \cos(\theta) + (Y \cdot u) 2c \sin(\theta) + mc^2$$

where $u \cdot v$ is the Euclidean dot product and $|u|$ is the Euclidean norm. To find our potential candidates for the minimum we set each partial derivative equal to zero and solve for θ and c.

$$\frac{\partial f}{\partial c} = 2(X \cdot u) \cos(\theta) + 2(Y \cdot u) \sin(\theta) + 2mc = 0$$

$$\frac{\partial f}{\partial \theta} = \sin(2\theta)(|Y|^2 - |X|^2) + 2(X \cdot Y) \cos(2\theta) \\ - 2c(X \cdot u) \sin(\theta) + 2c(Y \cdot u) \cos(\theta) = 0$$

From the first equation we get

$$c = -\frac{1}{m}((X \cdot u) \cos(\theta) + (Y \cdot u) \sin(\theta))$$

Plugging this expression for c into the second equation and simplifying using a few trigonometric identities yields

$$\sin(2\theta) \left[(|Y|^2 - |X|^2) - \frac{1}{m}((Y \cdot u)^2 - (X \cdot u)^2) \right] \\ + \cos(2\theta) \left[2(X \cdot Y) - \frac{2}{m}(X \cdot u)(Y \cdot u) \right] = 0$$

Assuming $\cos(2\theta) \neq 0$, rearranging and dividing by $\cos(2\theta)$ gives us

$$\tan(2\theta) = \frac{2((X \cdot u)(Y \cdot u) - m(X \cdot Y))}{m(|Y|^2 - |X|^2) - ((Y \cdot u)^2 - (X \cdot u)^2)}$$

Therefore we have that

$$\theta = \frac{1}{2} \arctan \left(\frac{2((X \cdot u)(Y \cdot u) - m(X \cdot Y))}{m(|Y|^2 - |X|^2) - ((Y \cdot u)^2 - (X \cdot u)^2)} \right)$$

Since $\tan(2\theta)$ has period $\frac{\pi}{2}$ we get two values θ and $\theta + \frac{\pi}{2}$. Now that we have θ, using the equation above we compute our c.

So given a set of m points in $\mathbb{R}^2$ the procedure for finding the line of best fit is as follows

(1) Compute θ_1, $\theta_2 = \theta_1 + \frac{\pi}{2}$ and their corresponding values c_1 and c_2.

(2) For each of the two possibilities, plug the values for θ_i, c_i into $f(\theta, c)$ and take the solution with the smaller value.
(3) Thus the best fit line is given by $x\cos(\theta_i) + y\sin(\theta_i) + c_i = 0$

True Error. If interested in computing the true error of the line of best fit, simply compute

$$\sum_{i=1}^{m} |x_i \cos(\theta) + y_i \sin(\theta) + c|$$

which gives the sum of the perpendicular distances from each point to the line.

4. Best Line Example

Take the set of points $S = \{(1,1),(2,1),(3,2),(6,4),(5,3),(4,4),(7,3)\}$ and let $X = (1,2,3,6,5,4,7)$, $Y = (1,1,2,4,3,4,3)$, and $u = (1,1,1,1,1,1,1)$ with $m = 7$.

Substituting into our formulas for θ and c gives us two sets of critical values

$$(\theta, c_1) \approx (0.478931, -4.73495)$$
$$(\theta + \frac{\pi}{2}, c_2) \approx (2.04973, -0.438791)$$

with errors (squared perpendicular distance) given by $f(\theta, c_1) \approx 34.7503$ and $f(\theta + \frac{\pi}{2}, c_2) \approx 2.964$. Thus our line of best fit is ℓ_2 and the equation of the other solution for θ is ℓ_1 given by

$$\ell_1 : -4.73495 + 0.887488x + 0.46083y = 0$$
$$\ell_2 : -0.438791 - 0.46083x + 0.887488y = 0$$

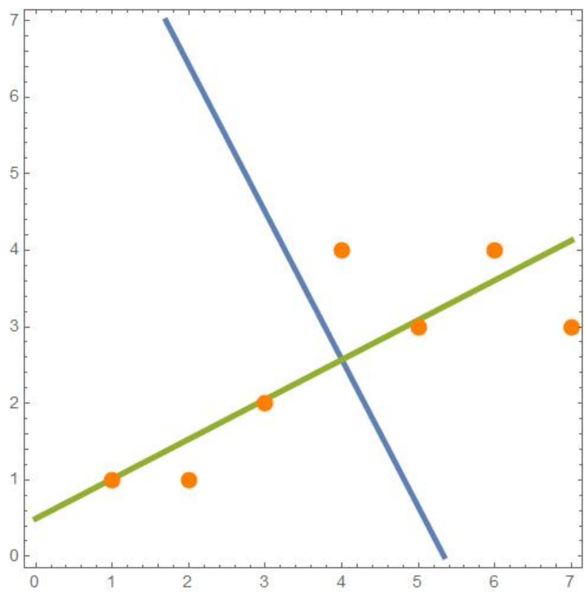

Figure 1. Blue: ℓ_1, Green: ℓ_2

5. Best Hyperplane

Notice in the best line example above we get two perpendicular lines that appear to evenly distribute the points into four quadrants. This can be generalized. For example, in the best plane example below with points in $\mathbb{R}^3$ we get three mutually orthogonal planes giving us eight octants that try to evenly distribute the points.

Let us assume that we have a collection of points $S = \{p_1, \dots, p_m\}$ in $\mathbb{R}^n$ and that we are interested in finding the best hyperplane $\Pi = \{(x_1, \dots, x_n) \in \mathbb{R}^n : a_1 x_1 + \cdots + a_n x_n + b = 0\}$ that fits these points. A natural way to select this plane is to find the one that minimize the sum of the perpendicular distances from the points to the plane. If we assume that the vector $a = (a_1, \dots, a_n)$ is a unit vector, then the distance from p_i to the plane Π is given by $|p_i \cdot a + b|$ where $u \cdot v$ is the Euclidean dot product. Therefore, the function to minimize is the function $\sum_{i=1}^m |p_i \cdot a + b|$. For convenience we will be minimizing the function

$$f(a,b) = \sum_{i=1}^m (p_i \cdot a + b)^2 \quad \text{subject to} \quad g(a,b) = a \cdot a = 1$$

Using the method of Lagrange multiplier, we get the possible minimums happen at points $(a, b, \lambda) \in \mathbb{R}^{n+1}$ such that $\nabla f = \lambda \nabla g$ and $g(a,b) = 1$. A direct computation shows that if $p_i = (x_{i1}, x_{i2}, \dots, x_{in})$ and we define

$$X_1 = (x_{11}, \dots, x_{m1}) \in \mathbb{R}^m, \ \dots, X_n = (x_{1n}, \dots, x_{mn}) \in \mathbb{R}^m \text{ and } u = (1, \dots, 1) \in \mathbb{R}^m$$

and the $n \times n$ matrix M with entry i,j given by the dot product $X_i \cdot X_j$, the vector $v \in \mathbb{R}^n$ with i^{th} entry $X_i \cdot u$, and L the matrix with entry i,j given by the product $v_i v_j$then we can rewrite f as follows

$$\begin{aligned} f(a,b) &= \sum_{i=1}^m (p_i \cdot a)^2 + 2b \sum_{i=1}^m (p_i \cdot a) + mb^2 \\ &= \sum_{i=1}^m \Big(\sum_{j=1}^n x_{ij} a_j\Big)^2 + 2b \sum_{i=1}^m \sum_{j=1}^n x_{ij} a_j + mb^2 \\ &= \sum_{i=1}^m \sum_{j=1}^n \sum_{k=1}^n x_{ij} a_j x_{ik} a_k + 2b \sum_{j=1}^n a_j \sum_{i=1}^m x_{ij} + mb^2 \\ &= a \cdot Ma + mb^2 + 2b\, a \cdot v \end{aligned}$$

From the expression above we get that the gradient of $f(a,b)$ is

$$\nabla f = (2Ma + 2bv, 2mb + 2a \cdot v)$$

Since $\nabla g = (2a, 0)$ then we can write the equations $\nabla f = \lambda \nabla g$ as

$$Ma + bv = \lambda a \quad \text{and} \quad mb + a \cdot v = 0$$

Therefore $b = -\frac{1}{m} a \cdot v$ and $Ka = \lambda a$ where $K = M - \frac{1}{m} L$.

So given a set of m points in $\mathbb{R}^n$ the procedure for finding the hyperplane of best fit is as follows

(1) Compute the matrices M, v, and L
(2) Compute the matrix K and its eigenvectors
(3) For each eigenvector a, compute the constant term b
(4) The elements of a are the coefficients of the hyperplane and b is the constant term $a_1x_1 + \cdots + a_nx_n + b = 0$
(5) Plugging each solution into the function $f(a,b)$ and taking the one that gives the smallest value is the plane of best fit

True Error. If interested in computing the true error of the hyperplane of best fit, simply compute

$$\sum_{i=1}^{m} \frac{|p_i \cdot a + b|}{\sqrt{\sum_{j=1}^{n} a_j^2}}$$

which gives the sum of the perpendicular distances from each point to the hyperplane.

6. Best Plane Example

Take the set of $m = 7$ points in $\mathbb{R}^3$ to be

$$S = \{(1,1,2),(1,2.5,1),(2,1.5,1),(3,2,1),(3,3,2),(4,3,2),(1,0.75,1)\}$$

A direct computation gives us

$$M = \begin{pmatrix} 41 & 34.25 & 23 \\ 34.25 & 32.0625 & 20.75 \\ 23 & 20.7 & 16 \end{pmatrix} \quad L = \begin{pmatrix} 225 & 206.25 & 150 \\ 206.25 & 189.063 & 137.5 \\ 150 & 137.5 & 100 \end{pmatrix}$$

$$K = \begin{pmatrix} 62/7 & 4.78571 & 11/7 \\ 4.78571 & 5.05357 & 1.10714 \\ 11/7 & 1.10714 & 12/7 \end{pmatrix} \quad v = \begin{pmatrix} 15 & 13.75 & 10 \end{pmatrix}$$

such that the eigenvalues and eigenvectors of K are

$$\begin{array}{ll} \xi_1 = 12.449 & a_1 = (-0.814171, -0.553232, -0.176239) \\ \xi_2 = 1.80811 & a_2 = (0.571409, -0.817313, -0.0741071) \\ \xi_3 = 1.36767 & a_3 = (-0.103044, -0.161041, 0.981554) \end{array}$$

Computing our constant terms b_i we get three candidates for the plane of best fit

$$\Pi_1 : 3.08313 - 0.814171x - 0.553232y - 0.176239z = 0$$
$$\Pi_2 : 0.486855 + 0.571409x - 0.817313y - 0.0741071z = 0$$
$$\Pi_3 : -0.865081 - 0.103044x - 0.161041y + 0.981554z = 0$$

with error (squared perpendicular distance) terms $f(a_1, b_1) \approx 12.4492$, $f(a_2, b_2) \approx 1.80811$ and $f(a_3, b_3) \approx 1.36767$. Therefore our plane of best fit is Π_3. See Figures below.

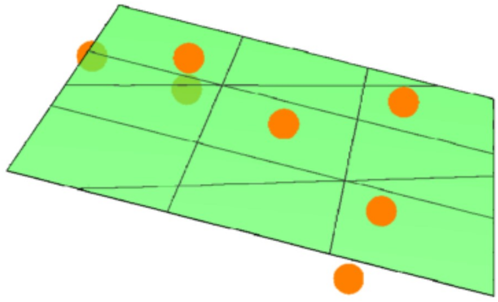

FIGURE 2. Plane of best fit Π_3 for S

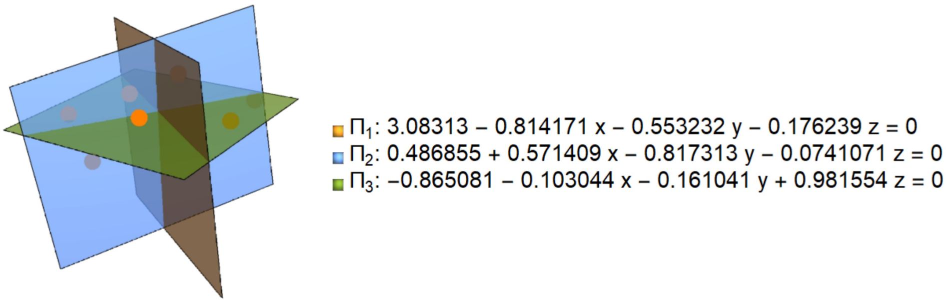

FIGURE 3. Π_1, Π_2, Π_3 divide $\mathbb{R}^3$ into octants. In this case Π_3 is the best plane

7. Best Ellipse

We will be considering three approaches for finding the best fitting ellipse given a set of points in $\mathbb{R}^2$. In the first approach we use a gradient descent algorithm such as the one included in the program Mathematica. In the second and third approach we solve it directly using partial derivatives and matrix algebra.

Our first approach using gradient descent requires us to find starting estimates for h, k, a, b, θ beforehand using relatively simple procedures in Mathematica.

For the second approach we take the equation for a general conic section and sum over all of our points, squaring each term. Minimizing this function is straight forward by taking partial derivatives and solving a system of linear equations.

In the third approach we perform some simple procedures in Mathematica to get an estimate on the angle the predicted major axis makes with the positive x-axis as well as the eccentricity. With these two extra pieces of information we perform a change of coordinates and remove the xy term from the equation of the conic, as well as force the eccentricity.

Since the orbits of planets are very close to perfect ellipses, all of our approaches will be assuming the equation of the conic represents an ellipse.

Approach 1. Recall that in the procedure for the best line and hyperplane we were able to use only the equation of the line and hyperplane. We follow a similar procedure here, in that we will use an equation of an ellipse and nothing else.

Let $S = \{p_1, \ldots, p_m\}$ be a collection of points in $\mathbb{R}^2$ under the standard basis. We define a new orthonormal basis given by $B = \{e'_1, e'_2\}$ where $e'_1 = (\cos(\theta), \sin(\theta))$ and $e'_2 = (-\sin(\theta), \cos(\theta))$. So for any point $p_i = (x_i, y_i)$ we have in the new coordinate system the point $(u_i, v_i) = (p_i \cdot e'_1, p_i \cdot e'_2)$. Furthermore, if $c = (h, k)$ under the standard basis, we take $(h_u, k_v) = (c \cdot e'_1, c \cdot e'_2)$

Thus the equation of a rotated ellipse given θ, a, b, h, k is given by

$$\frac{(u - h_u)^2}{a^2} + \frac{(v - k_v)^2}{b^2} - 1 = 0$$

where θ is the counterclockwise angle from the positive x direction. Notice that if a point p_i is on the ellipse then $\frac{(u_i-h_u)^2}{a^2} + \frac{(v_i-k_v)^2}{b^2} - 1 = 0$, otherwise $\mid \frac{(u_i-h_u)^2}{a^2} + \frac{(v_i-k_v)^2}{b^2} - 1 \mid > 0$. We will take this to be our measure of how close a point is to the ellipse. For convenience we will be minimizing the function

$$f(a, b, h, k, \theta) = \sum_{i=1}^{m} \left(\frac{(u_i - h_u)^2}{a^2} + \frac{(v_i - k_v)^2}{b^2} - 1 \right)^2$$

using Mathematica. There are two obvious issues with this approach. The first is that we may get stuck at a local minimum. The second is that the gradient descent algorithm might make a and b arbitrarily large. In the first example we will use the best line procedure to give us initial estimates for a, b, and θ.

Example of Approach 1. Coming soon.

Approach 2. Let $S = \{p_1, \ldots, p_m\}$ be a collection of points in $\mathbb{R}^2$. The equation for a generic conic can be written as

$$Ax^2 + Bxy + Cy^2 + Dx + Ey + F = 0$$

Taking the algebraic distance as our measure, we are interested in minimizing the function

$$f(A, B, C, D, E) = \sum_{i=1}^{m} (Ax_i^2 + Bx_iy_i + Cy_i^2 + Dx_i + Ey_i - 1)^2$$

Notice that we let $F = -1$. This is because we can always multiply through the equation by a constant term. However, this restricts our conic in that it cannot pass through the origin $(0, 0)$. Expanding f and taking partial derivatives yields

$$\frac{\partial f}{\partial A} = 2A(X_4Y_0) + 2B(X_3Y_1) + 2C(X_2Y_2) + 2D(X_3Y_0) + 2E(X_2Y_1) - 2(X_2Y_0)$$
$$\frac{\partial f}{\partial B} = 2A(X_3Y_1) + 2B(X_2Y_2) + 2C(X_1Y_3) + 2D(X_2Y_1) + 2E(X_1Y_2) + -2(X_1Y_1)$$
$$\frac{\partial f}{\partial C} = 2A(X_2Y_2) + 2B(X_1Y_3) + 2C(X_0Y_4) + 2D(X_1Y_2) + 2E(X_0Y_3) - 2(X_0Y_2)$$
$$\frac{\partial f}{\partial D} = 2A(X_3Y_0) + 2B(X_2Y_1) + 2C(X_1Y_2) + 2D(X_2Y_0) + 2E(X_1Y_1) - 2(X_1Y_0)$$
$$\frac{\partial f}{\partial E} = 2A(X_2Y_1) + 2B(X_1Y_2) + 2C(X_0Y_3) + 2D(X_1Y_1) + 2E(X_0Y_2) - 2(X_0Y_1)$$

such that $X_jY_k = \sum_{i=1}^{m} x_i^j y_i^k$.

Setting each equation equal to zero yields the matrix equation $Ax = b$ where

$$A = \begin{pmatrix} X_4Y_0 & X_3Y_1 & X_2Y_2 & X_3Y_0 & X_2Y_1 \\ X_3Y_1 & X_2Y_2 & X_1Y_3 & X_2Y_1 & X_1Y_2 \\ X_2Y_2 & X_1Y_3 & X_0Y_4 & X_1Y_2 & X_0Y_3 \\ X_3Y_0 & X_2Y_1 & X_1Y_2 & X_2Y_0 & X_1Y_1 \\ X_2Y_1 & X_1Y_2 & X_0Y_3 & X_1Y_1 & X_0Y_2 \end{pmatrix} \quad x = \begin{pmatrix} A \\ B \\ C \\ D \\ E \end{pmatrix} \quad b = \begin{pmatrix} X_2Y_0 \\ X_1Y_1 \\ X_0Y_2 \\ X_1Y_0 \\ X_0Y_1 \end{pmatrix}$$

Notice that A is a symmetric matrix. Assuming the $Det(A) \neq 0$ gives us a unique solution $x = A^{-1}b$.

Example of Approach 2. Take the set of $m = 7$ points in $\mathbb{R}^2$ to be

$$S = \{(2,1), (2,4), (3,1), (3,6), (4,2), (5,4), (5,6)\}$$

A direct computation gives us

$$A = \begin{pmatrix} 1700 & 1607 & 1765 & 384 & 365 \\ 1607 & 1765 & 2213 & 365 & 421 \\ 1765 & 2213 & 3122 & 421 & 570 \\ 384 & 365 & 421 & 92 & 89 \\ 365 & 421 & 570 & 89 & 110 \end{pmatrix} \quad b = \begin{pmatrix} 92 \\ 89 \\ 110 \\ 24 \\ 24 \end{pmatrix}$$

where $Det(A) = 57,566,592$ and therefore has a unique inverse. We leave the computation of A^{-1} to the reader. Therefore we have as a solution

$$x = A^{-1}b \approx \begin{pmatrix} -0.15008 \\ 0.0885364 \\ -0.0512472 \\ 0.684014 \\ 0.0894444 \end{pmatrix}$$

Thus the equation of our best fitting ellipse (see figure below) is given by

$$-0.15008x^2 + 0.0885364xy - 0.0512472y^2 + 0.684014x + 0.0894444y - 1 = 0$$

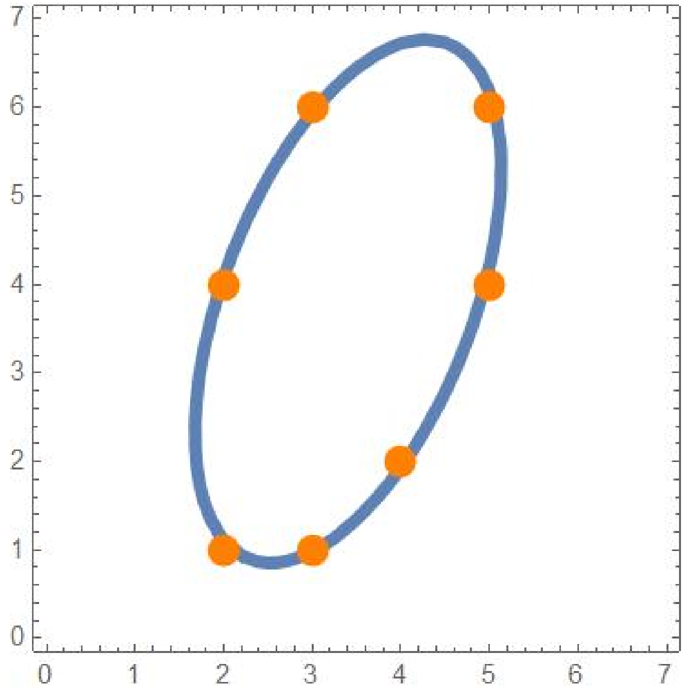

FIGURE 4

Approach 3. Assume we have a set of points in $\mathbb{R}^2$ that approximate an axis aligned ellipse with eccentricity e that does not contain the origin. Then a direct computation shows that we can look for an ellipse of the form

$$(1-e^2)Bx^2 + By^2 + Cx + Dy - 1 = 0$$

Using the same notation and following the same procedure as in approach 1 we have that $Ax = b$ where

$$A = \begin{pmatrix} X_4Y_0(1-e^2)^2 + 2X_2Y_2(1-e^2) + X_0Y_4 & X_3Y_0(1-e^2) + X_1Y_2 & X_2Y_1(1-e^2) + X_0Y_3 \\ X_3Y_0(1-e^2) + X_1Y_2 & X_2Y_0 & X_1Y_1 \\ X_2Y_1(1-e^2) + X_0Y_3 & X_1Y_1 & X_0Y_2 \end{pmatrix}$$

$$x = \begin{pmatrix} B \\ C \\ D \end{pmatrix} \quad b = \begin{pmatrix} X_2Y_0(1-e^2) + X_0Y_2 \\ X_1Y_0 \\ X_0Y_1 \end{pmatrix}$$

Notice that A is a symmetric matrix. Assuming the $Det(A) \neq 0$ gives us a unique solution $x = A^{-1}b$.

Example of Approach 3. While in a real application we will use other methods for computing the approximate eccentricity and angle the major axis makes with the positive x-axis, to illustrate Approach 3 we use our result from Approach 2 as a starting point. Using the equation we found we directly compute the eccentricity and angle the major axis of our ellipse makes with the positive x-axis giving us

$$e \approx 0.891353, \quad \theta \approx 1.205545$$

For each point $p_i \in S$ we perform a change of basis by computing $(p_i \cdot (\cos\theta, \sin\theta), p_i \cdot (-\sin\theta, \cos\theta))$ giving us

$$S' = \{(1.6484, -1.51088), (4.4505, -0.439331), (2.00559, -2.44492),$$
$$(6.67576, -0.658996), (3.2968, -3.02177), (5.52206, -3.24143), (7.39012, -2.52706)\}$$

A direct computation shows

$$A = \begin{pmatrix} 885.103 & 361.556 & -162.33 \\ 361.556 & 167.088 & -60.2855 \\ -162.33 & -60.2855 & 34.9117 \end{pmatrix} \quad b = \begin{pmatrix} 69.2468 \\ 30.9892 \\ -13.8444 \end{pmatrix}$$

where $Det(A) = 56113.6$ and therefore has a unique inverse. Thus the solution is

$$x = A^{-1}b \approx \begin{pmatrix} -0.167008 \\ 0.327863 \\ -0.606944 \end{pmatrix}$$

The equation of our best fitting ellipse is given by

$$-0.0343186x^2 + 0.327863x - 0.167008y^2 - 0.606944y - 1 = 0$$

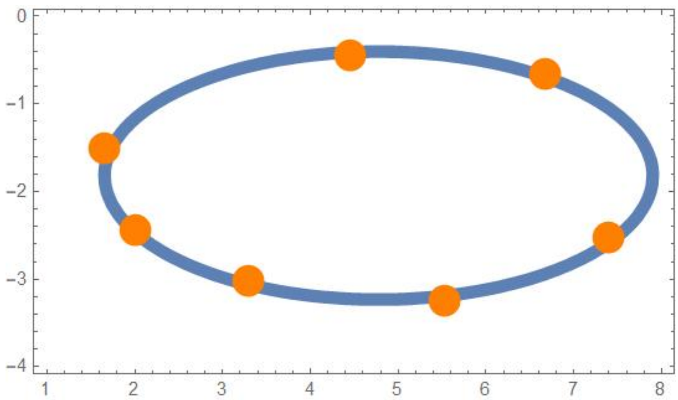

FIGURE 5

Note that if we wish to have the equation of this ellipse model our original (non-axis aligned) set of points, we simply use the angle θ found above and replace $x \to x\cos\theta + y\sin\theta$ and $y \to -x\sin\theta + y\cos\theta$ giving us the equation found above in the example of Approach 1

$$-0.15008x^2 + 0.0885364xy - 0.0512472y^2 + 0.684014x + 0.0894444y - 1 = 0$$

True Error. In order to determine how well our ellipse fits the set of points we use the method of Lagrange Multiplier. In particular we would like to find the point on the ellipse closest to our given point. Naturally the line segment joining these two points will be perpendicular to the ellipse. In order to accomplish this we use a program written in Mathematica.

Suppose we have an equation of an ellipse given by $g(x, y) = ax^2 + bxy + cy^2 + dx + ey - 1 = 0$. For each point (x_i, y_i) we would like to minimize the function $f(x, y) = (x - x_i)^2 + (y - y_i)^2$ subject to the constraint that

$g(x, y) = 0$. Using the method of Lagrange Multiplier we have $\nabla f = \lambda \nabla g$ which yields

$$\{2(x - x_i), 2(y - y_i)\} = \lambda\{d + 2ax + by, e + bx + 2cy\}$$

Solving for x and y in terms of λ gives

$$x = \frac{-2x_i - d\lambda - \frac{b\lambda(4y_i+2e\lambda+2bx_i\lambda-4ay_i\lambda+bd\lambda^2-2ae\lambda^2)}{4-4a\lambda-4c\lambda-b^2\lambda^2+4ac\lambda^2}}{2(-1 + a\lambda)}$$

$$y = \frac{4y_i + 2e\lambda + 2bx_i\lambda - 4ay_i\lambda + bd\lambda^2 - 2ae\lambda^2}{4 - 4a\lambda - 4c\lambda - b^2\lambda^2 + 4ac\lambda^2}$$

Now plugging these expressions for x and y into $g(x, y) = 0$ and simplifying yields a degree four polynomial in terms of λ. This is good news as we have a formula for solving for roots of quartic equations. We let Mathematica do the computation for us. We restrict ourselves to only the real solutions.

Once we have our λ's we compute our potential candidates (x, y) and see which choice provides the smallest value of $f(x, y)$. The point (x, y) which provides the smallest value of f is the closest point on the ellipse to our point (x_i, y_i). Therefore the minimum distance from (x_i, y_i) to our ellipse is $\sqrt{f(x, y)}$. Doing this procedure for each point (x_i, y_i) and taking the sum provides the true error of our best fitting ellipse.

Let us look at an example using Mathematica.

Example of Error. Suppose we have the ellipse

$$2x^2 - 2xy + y^2 + x - y - 1 = 0$$

and set of points

$$S = \{(.5, -.5), (.5, 1.5), (-1, 2), (1, 2.5), (-1.5, 0), (2, 1), (0, -2), (0, 1), (1, 0), (-.5, -.5)\}$$

Using the procedure outlined above in Mathematica we find that the respective minimum distances from each point to the ellipse are

$$\{0.295205, 0.421668, 0.99862, 0.446779, 0.477917, 0.96163, 1.11803, 0.406836, 0.423511, 0.421668\}$$

with a total error of 5.97187 (see Figure 6 below).

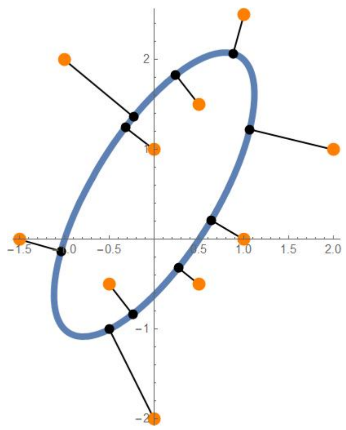

FIGURE 6. Closest points on an ellipse computing error

8. The Procedure for the Planets

First we explain the full procedure for finding the parametrization for Earth. In the next section we list the results for Mercury, Venus, Earth, Mars, and Jupiter.

Since we are finding equations to model relations, we must define our coordinate system that is being used. We use the settings given to us in the Horizons Web Interface. The origin of our coordinate system is taken to be the Solar System Barycenter. This is the center of mass of our solar system. Although it is always changing position, it can be thought of as being near the center of the sun.

The reference plane used is the ecliptic and mean equinox of reference epoch, and the reference system is ICRF/J2000.0 Documentation regarding reference frames and coordinate systems can be found on the Horizon documentation page `https://ssd.jpl.nasa.gov/?horizons_doc#frames`

Downloading and Importing Data. Navigating to the NASA HORIZONS Web-Interface at `https://ssd.jpl.nasa.gov/horizons.cgi` we download the data for the planet of interest and the Sun using the following settings

- Ephemeris Type: **Vectors**
- Target Body: [**Your Choice**]
- Coordinate Origin: **Solar System Barycenter** Time Span: Start=**2020-01-01**, Stop=**2020-12-31**, Step=**1 d**
- Table Settings: quantities code=**2**; output units=**KM-S**; CSV format=**YES**
- Display/Output: **download/save** (plain text file)

This provides us with two files (one for the sun, and the other for earth) containing the date of each observation, along with the x, y, z position coordinates, and vx, vy, vz velocity coordinates respectively.

We now consider the relative positions of the earth with respect to the sun taking $i = 1, 2, \ldots, 366$ giving us one full rotation around the sun.

$$p_i = (x, y, z)_{Earth} - (x, y, z)_{Sun}$$

For the planet results that follow, we pick dates that correspond with a single period/year, or one full rotation around the sun.

Plane of Best Fit and Projecting Points Into $\mathbb{R}^2$. With a table in Mathematica now containing our position vectors, we are able to use the procedure outlined in Section 5 to compute the plane of best fit $\Pi : ax + by + cz + d = 0$ with normal vector $n = (a, b, c)$.

Using our plane of best fit we map our points from $\mathbb{R}^3$ into $\mathbb{R}^2$ using the following procedure

(1) Using the normal vector $n = (a, b, c)$ from our plane of best fit, we consider the plane Π' with normal vector n such that Π' contains the origin. This plane is given by the equation $ax + by + cz = 0$

(2) Take the orthonormal basis for Π' given by

$$e_1 = \frac{(1,0,0) - ((1,0,0)\cdot n)n}{\|(1,0,0) - ((1,0,0)\cdot n)n\|}$$
$$e_2 = \frac{n \times e_1}{\|n \times e_1\|}$$

Notice that e_1 is the unit vector of the projection of $(1,0,0)$ into Π' and $Span\{e_1, e_2\} = \Pi'$

(3) We now project each point p_i onto our basis vectors e_1 and e_2 giving us a new set of coordinates given by

$$p_i' = (p_i \cdot e_1, p_i \cdot e_2) \in \mathbb{R}^2$$

Ellipse of Best Fit and Eccentricity. With our set of coordinates now in $\mathbb{R}^2$ we can use Approach 2 outlined in the best ellipse section. This gives us the equation

$$(1) \quad ax^2 + bxy + cy^2 + dx + ey - 1 = 0$$

We will now compute the parametrization and eccentricity of (1). To do this we define two new basis vectors

$$E1 = (\cos\theta, \sin\theta) \quad \text{and} \quad E2 = (-\sin\theta, \cos\theta)$$

which form an orthonormal basis of $\mathbb{R}^2$. To convert between a point (x, y) under the standard basis, to the equivalent point (u, v) under the new basis $\{E1, E2\}$ we have the set of relations

$$u = x\cos\theta + y\sin\theta$$
$$v = -x\sin\theta + y\cos\theta$$

$$x = u\cos\theta - v\sin\theta$$
$$y = u\sin\theta + v\cos\theta$$

Making the substitution for x and y into equation (1) gives us

$$(2) \quad 0 = -1 + e(v\cos\theta + u\sin\theta) + c(v\cos\theta + u\sin\theta)^2$$
$$+ d(u\cos\theta - v\sin\theta) + b(v\cos\theta + u\sin\theta)(u\cos\theta - v\sin\theta)$$
$$+ a(u\cos\theta - v\sin\theta)^2$$

Expanding and collecting the coefficient of the uv term gives us

Coefficient of uv term: $\quad b\cos^2\theta - 2a\cos\theta\sin\theta + 2c\cos\theta\sin\theta - b\sin^2\theta$

Setting this equation equal to zero and solving for θ yields

$$\theta = \frac{1}{2}\arctan\left(\frac{b}{a-c}\right)$$

assuming $\cos(2\theta) \neq 0$.

Substituting our values for a, b, c from (1) into this equation for θ, and substituting θ back into (2) we end up with a new equation for the ellipse with no uv term, and thus its major and minor axes are parallel to the basis vectors $E1$ and $E2$. Note the θ found above is the θ we use in our orthonormal basis $\{E1, E2\}$ which defines our coordinates (u, v).

Using the notation a', b', d', e' to be our new coefficients after making the substitution for θ into (2) we have

$$a'v^2 + d'v + b'u^2 + e'u = 0$$

Completing the square gives us the equation of the ellipse

$$\frac{\left(u + \frac{e'}{2b'}\right)^2}{\frac{1}{b'}\left(\frac{d'^2}{4a'} + \frac{e'^2}{4b'}\right)} + \frac{\left(v + \frac{d'}{2a'}\right)^2}{\frac{1}{a'}\left(\frac{d'^2}{4a'} + \frac{e'^2}{4b'}\right)} = 1$$

Therefore the parametrization in terms of $t \in [0, 2\pi)$ is

$$\alpha(t) = (h, k) + r\cos t(\cos\theta, \sin\theta) + s\sin t(-\sin\theta, \cos\theta)$$

where θ is the same θ found above and

$$h = (\frac{-e'}{2b'}, \frac{-d'}{2a'}) \cdot (\cos\theta, -\sin\theta)$$
$$k = (\frac{-e'}{2b'}, \frac{-d'}{2a'}) \cdot (\sin\theta, \cos\theta)$$
$$r = \sqrt{\frac{1}{b'}\left(\frac{d'^2}{4a'} + \frac{e'^2}{4b'}\right)}$$
$$s = \sqrt{\frac{1}{a'}\left(\frac{d'^2}{4a'} + \frac{e'^2}{4b'}\right)}$$

Furthermore the eccentricity is given by

$$\frac{\sqrt{s^2 - r^2}}{s} \quad \text{or} \frac{\sqrt{r^2 - s^2}}{r}$$

depending on whether $s > r$ or $s \leq r$. We now take this parametrization $\alpha(t)$ in $\mathbb{R}^2$ and put it back into $\mathbb{R}^3$ onto its original position in the plane of best fit.

Recall that our plane of best fit is $\Pi : ax + by + cz + d = 0$ with normal vector $n = (a, b, c)$. We would like to translate our points from $\Pi' : ax+by+cz = 0$ back to Π. To do this we need to find the point on Π that corresponds to the origin of Π'. In other words, where the vector n intersects Π.

Letting $p_0 = tn = t(a, b, c)$ for some $t \in \mathbb{R}$ and substituting p_0 into the equation for Π gives us $(a^2 + b^2 + c^2)t + d = 0$ and since n is a unit vector $t = -d$. Thus $p_0 = -dn = -d(a, b, c)$.

Therefore the parametrization of the best fitting ellipse through our original set of points $p_i \in \mathbb{R}^3$ is given by

$$\phi(t) = p_0 + \eta_1 E_1 + \eta_2 E_2$$

where E_1, E_2 are defined as above and

$$\eta_1 = h + r\cos t\cos\theta - s\sin t\sin\theta$$
$$\eta_2 = k + r\cos t\sin\theta + s\sin t\cos\theta$$

Notice that η_1, η_2 are the first and second entries of $\alpha(t)$ taken from the equation above.

We now show a full example using a small set of points in $\mathbb{R}^3$.

Example of Procedure. Take the set of $m = 10$ points in $\mathbb{R}^3$ to be

$$S = \{(2, 3.37, 3.45), (0.3, 2.07, 2.26), (0.61, 1.29, -0.27), (1.24, 1.17, -0.55), (2.99, 2.04, 0.95), (3.48, 3, 3.56), (2.58, 3.8, 2.89), (0.14, 1.49, 1.48), (0.06, 0.56, 0.96), (0.64, 0.57, 0.62)\}$$

Using the procedure for the plane of best fit we find

$$\Pi : 0.332541x - 0.840721y + 0.427323z + 0.504806 = 0$$

with $n = (0.332541, -0.840721, 0.427323)$ and $d = 0.504806$

Now letting Π' be the parallel plane through the origin, we compute basis vectors

$$E1 = (0.943089, 0.296445, -0.150678)$$
$$E2 = (1.02198 \cdot 10^{-17}, 0.453111, 0.891454)$$

Projecting our set of points onto the basis vectors and using our best ellipse procedure gives us

$$\alpha(t) = (2.05437, 2.49305) + 2.34605 \cos t(\cos u, \sin u) + 1.5626 \sin t(-\sin u, \cos u)$$

where $u = -2.02649$ and $t \in [0, 2\pi)$.

Note that $\alpha(t) \in \mathbb{R}^2$ so our final step is to put it back onto the plane of best fit Π through the original set of points in $\mathbb{R}^3$.

Taking

$$p_0 = (-0.167869, 0.424401, -0.215715)$$
$$\eta_1 = 2.05437 - 1.03246 \cos t + 1.40315 \sin t$$
$$\eta_2 = 2.49305 - 2.10665 \cos t - 0.687677 \sin t$$

where $p_0 = -dn$ and η_1, η_2 are the first and second entries of $\alpha(t)$ gives us

$$\phi(t) = p_0 + \eta_1 E_1 + \eta_2 E_2 \in \mathbb{R}^3 \quad \text{for} \quad t \in [0, 2\pi)$$

as our ellipse of best fit through the set of points p_i projected onto the plane Π. See Figure 7 below.

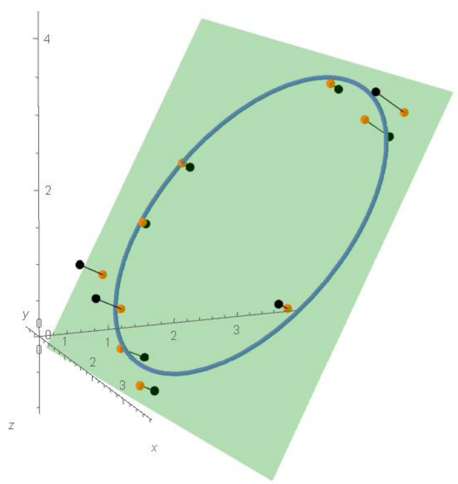

FIGURE 7. Image showing our plane Π, ellipse ϕ, and points (black) p_i with their projections (orange)

9. The Results for the Planets

The number of data points used for each planet is the number of days in one period (orbit around the sun). The date range of the data used is given next to the planet name. All results are in kilometers.

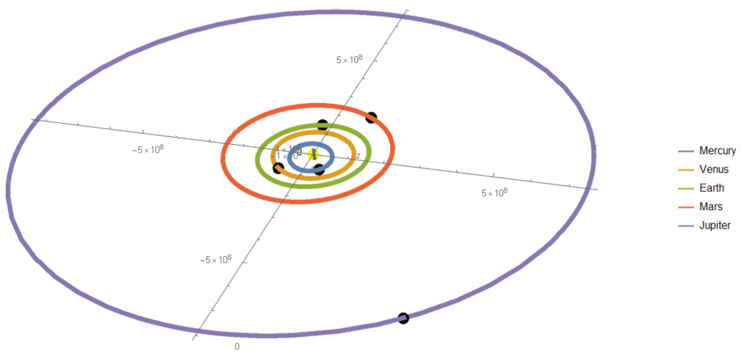

Figure 8. Angled view: Positions on January 1, 2021

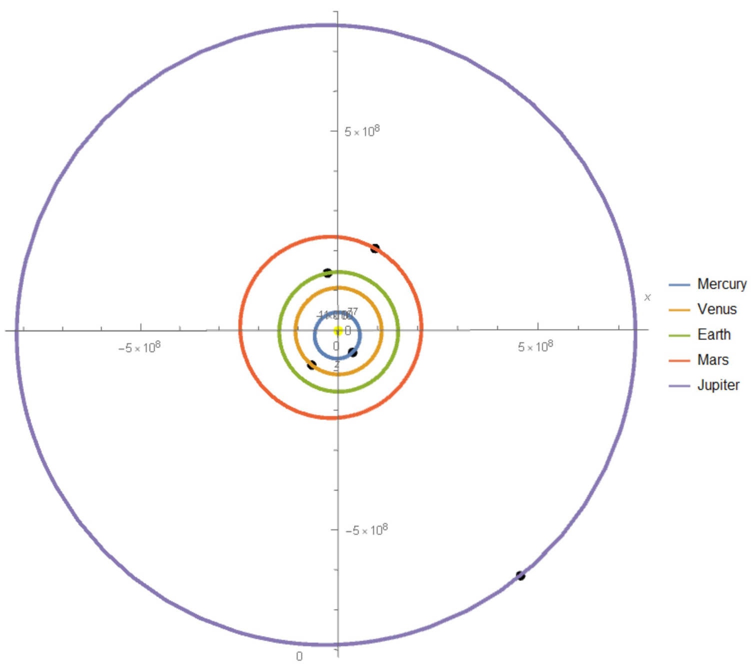

Figure 9. Top view: Positions on January 1, 2021

Mercury (2021.01.01 - 2021.03.29)

Semi-major axis	$5.7909 \cdot 10^7$ km
Semi-minor axis	$5.66714 \cdot 10^7$ km
Eccentricity	0.205633
Parametrization	$\phi(t) = p_0 + \eta_1 E_1 + \eta_2 E_2 \in \mathbb{R}^3 \quad \text{for} \quad t \in [0, 2\pi)$
$E1 = (0.995847, 0.00741525, -0.0907438)$	
$E2 = (1.17303 \cdot 10^{-18}, 0.996678, 0.0814449)$	
$p_0 = (-0.552638, 0.492306, -6.02457)$	
$\eta_1 = -2.62244 \cdot 10^6 + 5.52795 \cdot 10^7 \cos t + 1.27559 \cdot 10^7 \sin t$	
$\eta_2 = -1.16159 \cdot 10^7 - 1.24833 \cdot 10^7 \cos t + 5.64866 \cdot 10^7 \sin t$	

Venus (2020.01.01 - 2020.08.12)

Semi-major axis	$1.08211 \cdot 10^8$ km
Semi-minor axis	$1.08206 \cdot 10^8$ km
Eccentricity	0.00946215
Parametrization	$\phi(t) = p_0 + \eta_1 E_1 + \eta_2 E_2 \in \mathbb{R}^3 \quad \text{for} \quad t \in [0, 2\pi)$
$E1 = (0.998339, 0.000790377, -0.0576002)$	
$E2 = (-3.64763 \cdot 10^{-20}, 0.999906, 0.0137205)$	
$p_0 = (-1.55863, 0.370618, -27.0095)$	
$\eta_1 = 484649. + 8.14207 \cdot 10^7 \cos t + 7.12724 \cdot 10^7 \sin t$	
$\eta_2 = -548343. - 7.12756 \cdot 10^7 \cos t + 8.14171 \cdot 10^7 \sin t$	

Earth (2020.01.01 - 2020.12.31)

Semi-major axis	$1.49597 \cdot 10^8$ km
Semi-minor axis	$1.49578 \cdot 10^8$ km
Eccentricity	0.0160063
Parametrization	$\phi(t) = p_0 + \eta_1 E_1 + \eta_2 E_2 \in \mathbb{R}^3 \quad \text{for} \quad t \in [0, 2\pi)$
$E1 = (1., -1.29884 \cdot 10^{-10}, -2.80132 \cdot 10^{-6})$	
$E2 = (-1.93088 \cdot 10^{-27}, 1., -0.0000463654)$	
$p_0 = (0.0000523882, 0.000867092, 18.7013)$	
$\eta_1 = 562934. + 1.47953 \cdot 10^8 \cos t - 2.19871 \cdot 10^7 \sin t$	
$\eta_2 = -2.43885 \cdot 10^6 + 2.19843 \cdot 10^7 \cos t + 1.47972 \cdot 10^8 \sin t$	

Mars (2019.01.01 - 2020.11.17)

Semi-major axis	$2.27947 \cdot 10^8$ km
Semi-minor axis	$2.26938 \cdot 10^8$ km
Eccentricity	0.0939591
Parametrization	$\phi(t) = p_0 + \eta_1 E_1 + \eta_2 E_2 \in \mathbb{R}^3 \quad \text{for} \quad t \in [0, 2\pi)$
$E1 = (0.999699, 0.0005137, -0.0245161)$	
$E2 = (1.17962 \cdot 10^{-19}, 0.999781, 0.020949)$	
$p_0 = (5.00288, -4.27273, 203.914)$	
$\eta_1 = -1.94744 \cdot 10^7 + 2.09228 \cdot 10^8 \cos t + 9.00624 \cdot 10^7 \sin t$	
$\eta_2 = 8.61113 \cdot 10^6 - 9.04626 \cdot 10^7 \cos t + 2.08302 \cdot 10^8 \sin t$	

Jupiter (2009.01.01 - 2020.11.09)

Semi-major axis	$7.78286 \cdot 10^8$ km
Semi-minor axis	$7.77322 \cdot 10^8$ km
Eccentricity	0.049781
Parametrization	$\phi(t) = p_0 + \eta_1 E_1 + \eta_2 E_2 \in \mathbb{R}^3 \quad \text{for} \quad t \in [0, 2\pi)$
$E1 = (0.99975, -0.0000929091, -0.0223711)$	
$E2 = (1.0625 \cdot 10^{-20}, 0.999991, -0.00415306)$	
$p_0 = (-8.40464, -1.55987, -375.592)$	
$\eta_1 = -3.68041 \cdot 10^7 + 7.63376 \cdot 10^8 \cos t - 1.51425 \cdot 10^8 \sin t$	
$\eta_2 = -9.41362 \cdot 10^6 + 1.51613 \cdot 10^8 \cos t + 7.6243 \cdot 10^8 \sin t$	