ASCOT AND COMPACT
TOOLS FOR ESTIMATING SOFTWARE DEVELOPMENT RESOURCES AND CUBESAT MISSION COSTS

Christian Smart[1] (Presenter), Michael DiNicola[1] (Presenter), Sherry Stukes[1], Michael Saing[1], Joseph Mrozinski[1], Takuto Ishimatsu[1], Shannon Statham[1], Vicky Nilsen[2]

April 24, 2024

[1]Jet Propulsion Laboratory, California Institute of Technology
[2]National Aeronautics and Space Administration Headquarters

OVERVIEW

New Task Managers looking forward to working with you!

ONSET Repository

Common Techniques used by ASCoT and COMPACT

Short dive into ASCoT

Short dive into COMPACT

Christian Smart
ASCoT Task Manager

- One year at JPL conducting independent cost estimates and JCLs to support milestone reviews
- Author of *Solving for Project Risk Management*
- Cost estimator since 1999
- PhD in Applied Mathematics

Mike DiNicola
COMPACT Task Manager

- Going on 17 years at JPL developing probabilistic risk models to assess cost and science & engineering requirements
- Lead statistician for NICM since 2014
- Works with JPL's biotech group to assess planetary protection methods
- BS & MA in Pure Mathematics

ONSET ARCHITECTURE

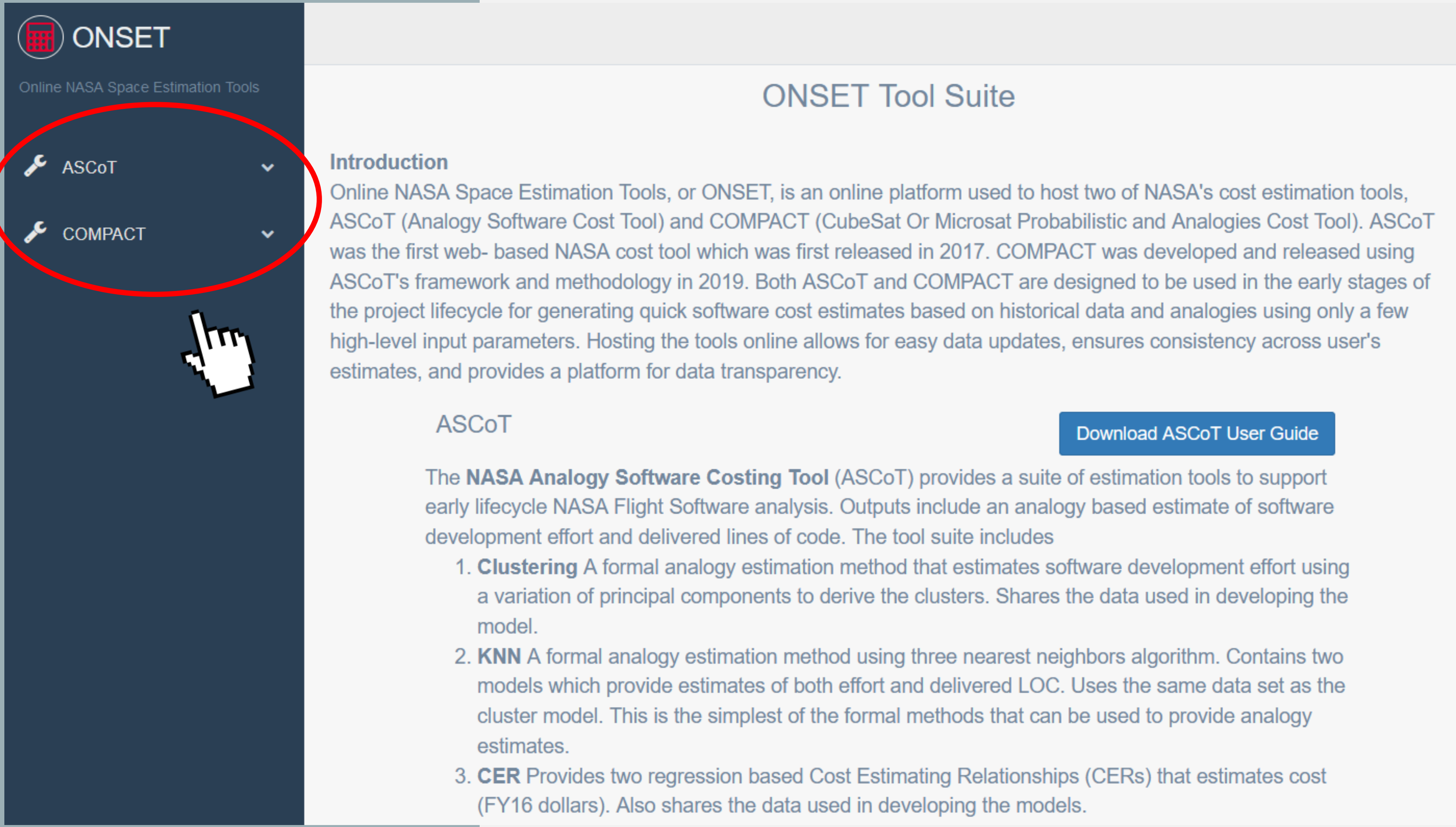
Two tools currently hosted on ONSET:

ASCoT
(Analogy Software Cost Tool)

COMPACT
(CubeSat Or Microsat Probabilistic and Analogies Cost Tool)

ASCoT and COMPACT share algorithms, team members, and philosophies.

Both tools are hosted online in ONSET at https://onset.jpl.nasa.gov/



ONSET

Online NASA Space Estimation Tools

ASCoT

COMPACT

ONSET Tool Suite

Introduction

Online NASA Space Estimation Tools, or ONSET, is an online platform used to host two of NASA's cost estimation tools, ASCoT (Analogy Software Cost Tool) and COMPACT (CubeSat Or Microsat Probabilistic and Analogies Cost Tool). ASCoT was the first web- based NASA cost tool which was first released in 2017. COMPACT was developed and released using ASCoT's framework and methodology in 2019. Both ASCoT and COMPACT are designed to be used in the early stages of the project lifecycle for generating quick software cost estimates based on historical data and analogies using only a few high-level input parameters. Hosting the tools online allows for easy data updates, ensures consistency across user's estimates, and provides a platform for data transparency.

ASCoT

Download ASCoT User Guide

The **NASA Analogy Software Costing Tool** (ASCoT) provides a suite of estimation tools to support early lifecycle NASA Flight Software analysis. Outputs include an analogy based estimate of software development effort and delivered lines of code. The tool suite includes

1. **Clustering** A formal analogy estimation method that estimates software development effort using a variation of principal components to derive the clusters. Shares the data used in developing the model.
2. **KNN** A formal analogy estimation method using three nearest neighbors algorithm. Contains two models which provide estimates of both effort and delivered LOC. Uses the same data set as the cluster model. This is the simplest of the formal methods that can be used to provide analogy estimates.
3. **CER** Provides two regression based Cost Estimating Relationships (CERs) that estimates cost (FY16 dollars). Also shares the data used in developing the models.

COMMON ESTIMATING TECHNIQUES

Formalization of analogic cost estimation process using Euclidean distance as "similarity" metric

Intended for early-on, ballpark cost estimates

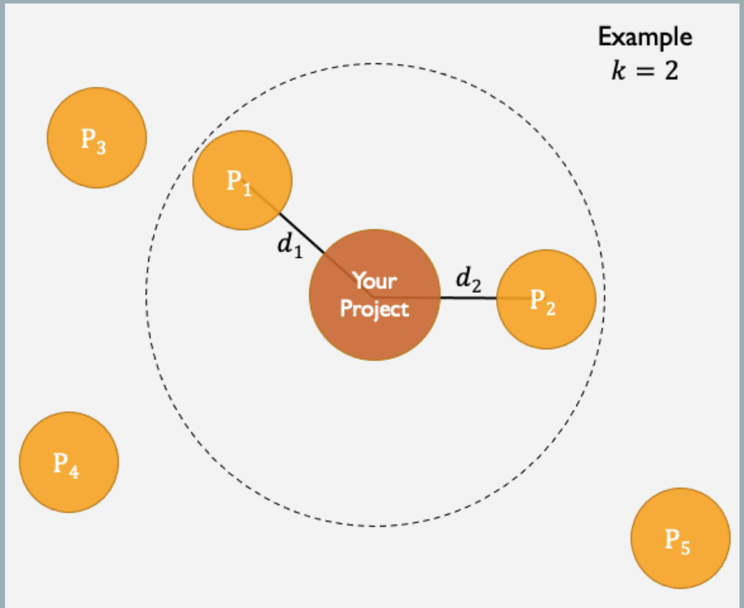
Cost estimate is a weighted average of the k most similar missions

Weighted by the inverse of the distance (i.e. closer missions have more weight)

Returns the nearest neighbors for analogy purposes

Users can judge whether analogues are appropriate

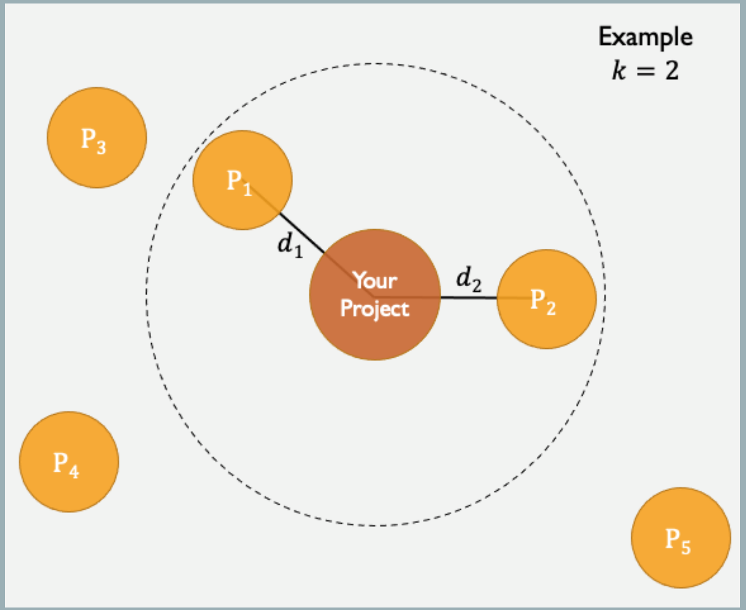
How does k-Nearest Neighbors (kNN) work?



$$\text{Cost(Your Project)} = \frac{\dfrac{\text{Cost}(\text{P}_1)}{d_1} + \dfrac{\text{Cost}(\text{P}_2)}{d_2}}{\dfrac{1}{d_1} + \dfrac{1}{d_2}}$$

ASCOT
(ANALOGY SOFTWARE COST TOOL)

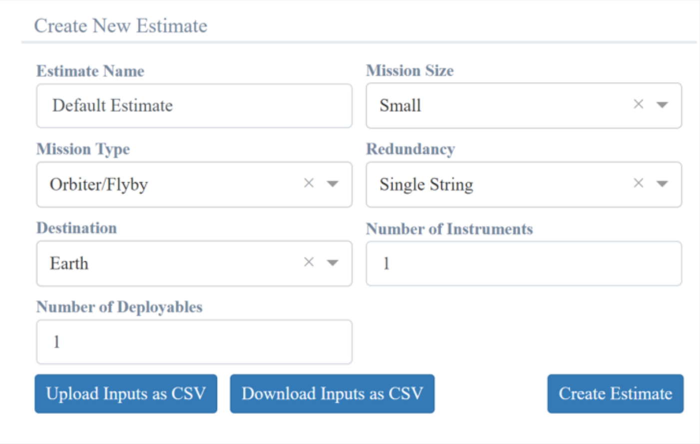
How does k-Nearest Neighbors (kNN) work?



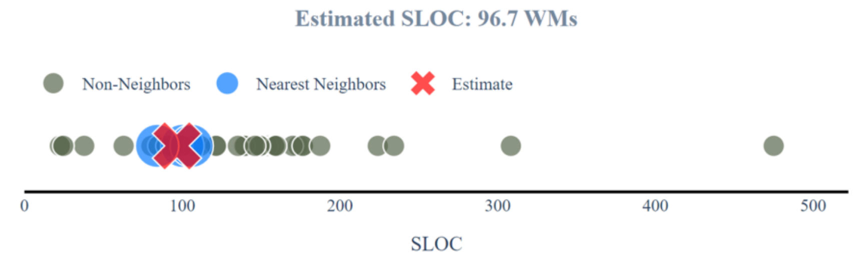
$$\text{Cost(Your Project)} = \frac{\frac{\text{Cost}(P_1)}{d_1} + \frac{\text{Cost}(P_2)}{d_2}}{\frac{1}{d_1} + \frac{1}{d_2}}$$

- Estimates development cost effort and SLOC for flight software using a variety of techniques
  - Analogic cost estimation via k-Nearest Neighbors (kNN) and clustering
  - Simple Bayesian CERs
  - Probabilistic version of USC's Constructive Cost Model (COCOMO II)
- Data: 51 Missions

Inputs



Outputs



Neighbors	Distance	SLOC
WISE	1	100.1
OCO-1	1	106
IRIS	1	84.1

ASCOT DATABASE

Database Status

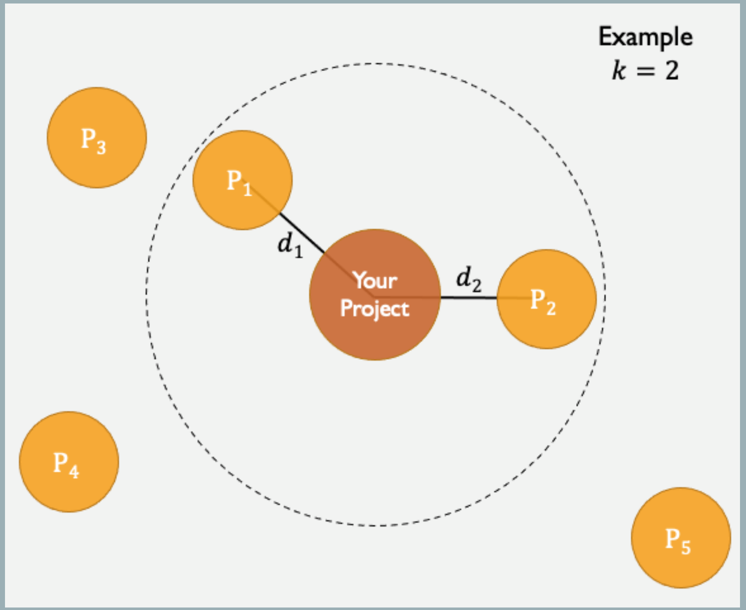
Ongoing Data Collection Efforts

Mission Name	Effort (WMs)	Inheritance	Size	Type	Redundancy	Destination	# Instruments	# Deployables
Cassini	3291.8	Very Low To None	Very Large	Orbiter/ Flyby	Dual String - Warm backup	Outer	12	4
GLL (Galileo)	1170	Very Low To None	Very Large	Orbiter/ Flyby	Dual String - Warm backup	Outer	11	8
MER	1833.3	Low	Very Large	Rover	Dual String - Warm backup	Inner	5	7
MPF	1080	Very Low To None	Medium	Rover	Single String	Inner	3	10
MSL	1910	Very Low To None	Very Large	Rover	Dual String - Warm backup	Inner	10	6
Insight	803	Very High	Large	Lander	Dual String - Warm backup	Inner	5	2
Phoenix	616.5	High	Medium	Lander	Dual String - Warm backup	Inner	4	3
Dawn	573	Very High	Medium	Orbiter/ Flyby	Dual String - Cold backup	Astr/Com	3	1
GRAIL	329	Very High	Medium	Orbiter/ Flyby	Single String	Inner	4	1
JUNO	425	High	Large	Orbiter/ Flyby	Dual String - Cold backup	Outer	9	1
Kepler	446	Medium	Medium	Observatory	Dual String - Cold backup	Inner	4	0
LADEE	451.6	Very High	Medium	Orbiter/ Flyby	Single String	Inner	4	0
MAVEN	621.3	Very High	Medium	Orbiter/ Flyby	Dual String - Cold backup	Inner	8	1
MRO	691	Very High	Large	Orbiter/ Flyby	Dual String - Cold backup	Inner	7	2
Messenger	534.9	Medium	Medium	Orbiter/ Flyby	Dual String - Cold backup	Inner	7	2

ASCOT'S KNN AND CLUSTERING ALGORITHMS USE PCA

- Intended for cost estimates early in the lifecycle of a project
- May provide more accurate estimates than parametrics when data are sparse or noisy
- Cost estimate is a weighted average of either the k most similar missions or the missions within the same cluster
    - Cost drivers: Mission Size, Redundancy, Inheritance, Mission Type, Destination, # of Instruments, & # of Deployables
    - Cost drivers are correlated, so we use Principal Components Analysis (PCA) to calculate similarity
    - Weighted by the inverse of the distance (i.e. closer missions have more weight)

How does k-Nearest Neighbors (kNN) work?



$$\text{Cost(Your Project)} = \frac{\frac{\text{Cost}(P_1)}{d_1} + \frac{\text{Cost}(P_2)}{d_2}}{\frac{1}{d_1} + \frac{1}{d_2}}$$

ASCOT BAYESIAN CER

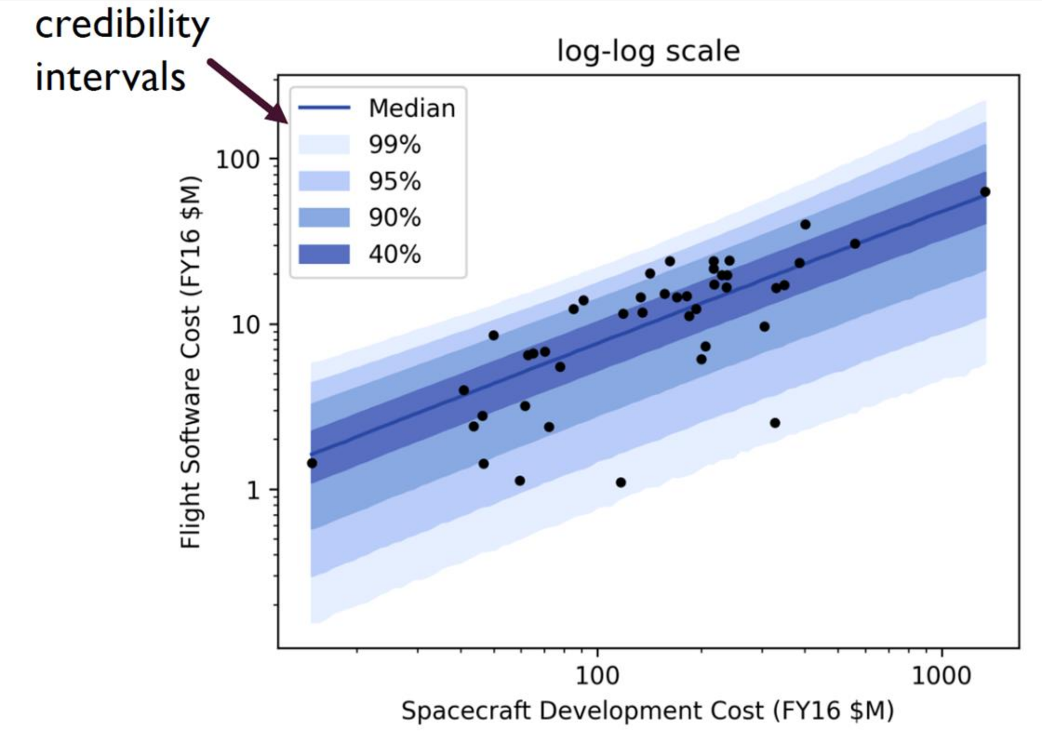
Single-variable regression – co-variate is total spacecraft development cost, measured in $

Single-variable regression performs better out-of-sample than multivariate CERs due to overfitting

Bayesian approach allows for consideration of experience and expert judgment in setting the priors

Both parameter uncertainty and uncertainty about the regression are incorporated

Regression model is a power equation and the skew normal is used to model regression error (influenced less by low cost outliers than the lognormal)



$$\log(\text{Software Cost}) = \beta_0 + \beta_1 \log(\text{Spacecraft Cost}) + \epsilon$$
$$\epsilon \sim \text{SkewNormal}(\sigma, \alpha)$$

COCOMO II

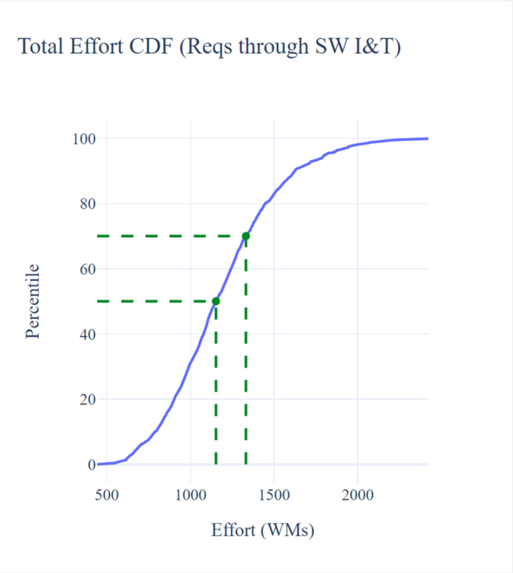
The COnstructive COst MOdel (COCOMO) was initially developed by Barry Boehm at USC in the late 1970s

COCOMO II was developed in the mid-1990s

Database consists of 161 projects and has 23 inputs

ONSET provides a probabilistic implementation of COCOMO II that allows users to input low, most likely, and high values for all 23 parameters

User can select 100 – 10,000 iterations for a Monte Carlo simulation



KNN ANALOGY EXAMPLE

Develop an estimate for the software development effort using kNN for the proposed DragonStar project

Inputs:

Estimate Name	**Inheritance Level**
Dragon Star	Low
Mission Size	**Mission Type**
Small	Orbiter/Flyby
Redundancy	**Destination**
Single String	Asteroid/Comet
Number of Instruments	**Number of Deployables**
2	1

Estimated Effort: 936.9 WMs

Neighbors	Distance	Effort (WMs)
DS1	1	1042.8
Deep_Impact	2.646	1047.9
Stardust	2.646	545.5

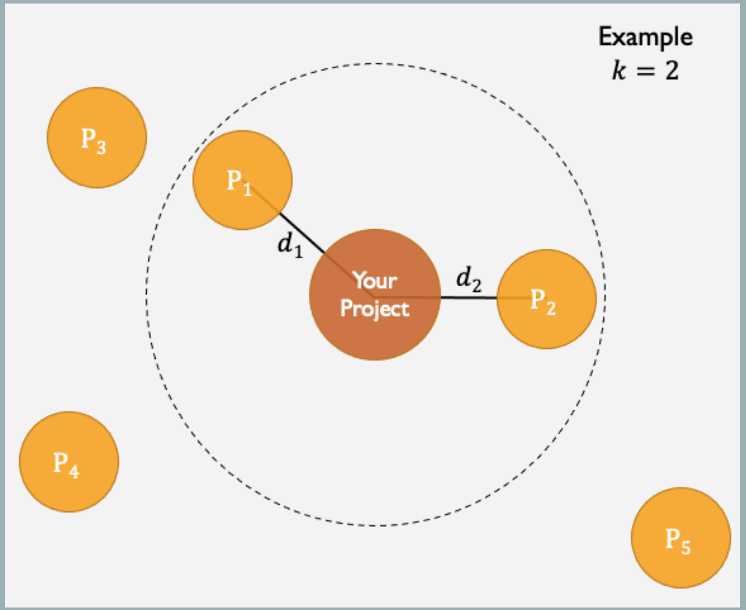
Estimated SLOC: 140.2 WMs

Neighbors	Distance	SLOC
DS1	0	140.2
Dawn	1.732	160.9
WISE	2.449	100.1
Contour	2.449	37.9
OCO-1	2.449	106
EO1	2.449	150
Stardust	2.449	100
IRIS	2.449	84.1
Glory	2.449	109.4
GRAIL	2.449	173.4

COMPACT
(CUBESAT OR MICROSAT PROBABILISTIC AND ANALOGIES COST TOOL)

"How much will my CubeSat project cost?"

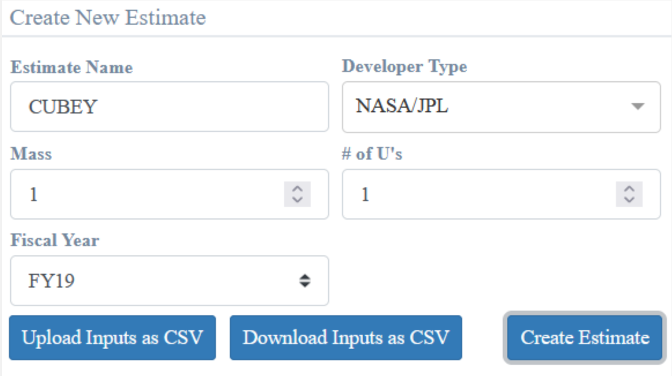
How does k-Nearest Neighbors (kNN) work?



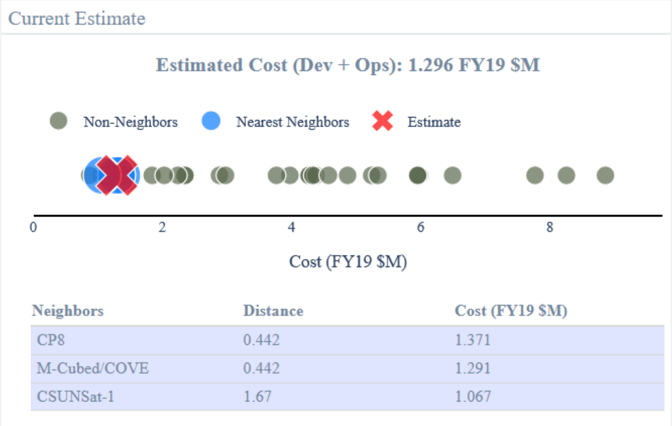
$$\text{Cost(Your Project)} = \frac{\frac{\text{Cost}(P_1)}{d_1} + \frac{\text{Cost}(P_2)}{d_2}}{\frac{1}{d_1} + \frac{1}{d_2}}$$

- Analogic cost estimation using k-Nearest Neighbors (kNN)
- Estimates full-lifecycle cost (dev + ops) of an input project based on a weighted average cost of the k most similar completed CubeSat projects
- Data: JPL & NASA affiliated CubeSat projects (N=40)
- We also have CERs for those who are not allowed access to data

Inputs



Outputs



COMPACT DATABASE

Database includes normalized cost and technical data from 40 CubeSat missions (6 more data points added in the last year)

Ongoing Data Collection Efforts include:

BioSentinel	CuPID	CuSP
CUTE	Lunah-Map	Lunar Flashlight
Lunar IceCube	LunIR	NEA Scout
Pan	petit-Sat	

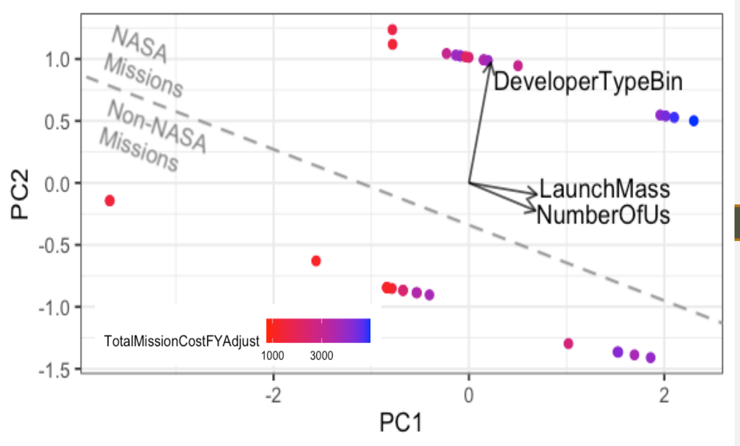
CubeSat Name	
# of U's	
Total Mass (kg) per CubeSat	
# of CubeSats developed/launched	
Peak Power (W) per CubeSat (provide draw and capability, if possible)	
Average Power (W) per CubeSat (specify orbit average or nominal)	
# of Science Instruments per CubeSat	
Design Life (Months from launch to end of primary mission)	
Total Development Schedule (Months from ATP to Pre-ship review)	
Total Mission Cost (Development + Operations)	
Fiscal Year of Cost	
Primary Developer	
Sponsor Organization and/or Partnering Organizations	
NASA Implementation Type (7120.5 / 7120.8 / DNH)	
CubeSat Website	
Operational Status/Mission Success?	

Summary Data Collection Form

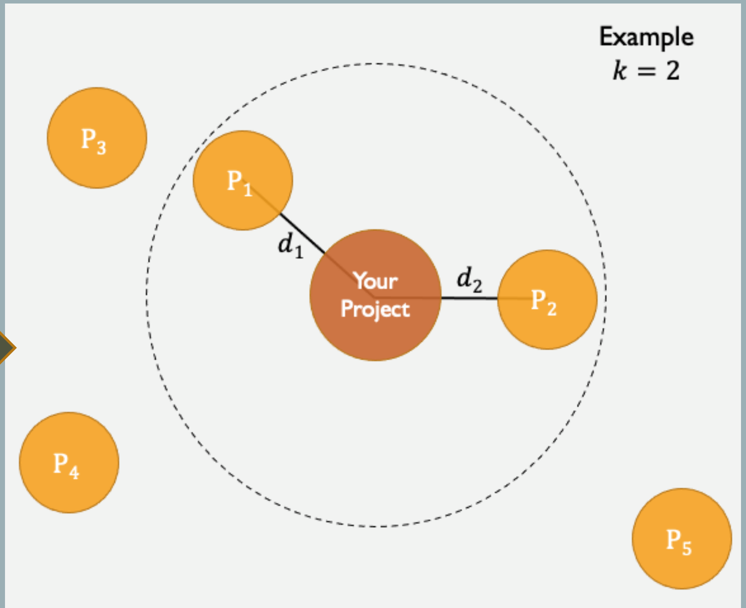
Mission Name	Launch Date	Mission Type	Developer Type	Mass (kg)	# of Us	# of Flight Units
CTIM-FD	7/2/2022	Tech Demo	Other	10.6	6	1
CIRiS-BATC	12/5/2019	Tech Demo	NASA	14	6	1
MiniCarb	12/5/2019	Science	NASA	6.4	6	1
E-TBEx	6/25/2019	Science	Hybrid	3.8	3	2
Kenobi	4/17/2019	Tech Demo	NASA	2.1	3	1
Seeker	4/17/2019	Tech Demo	NASA	4.2	3	1
AIBus	12/16/2018	Tech Demo	NASA	4	3	1
CeReS	12/16/2018	Science	NASA	4.3	3	1
CubeSail	12/16/2018	Tech Demo	Hybrid	1.7	1.5	2
Shields-1	12/16/2018	Tech Demo	NASA	6.9	3	1
STF-1	12/16/2018	Tech Demo	Hybrid	3.0	3	1
CSIM-FD	12/3/2018	Tech Demo	NASA	10.2	6	1
MinXSS-2	12/3/2018	Science	Hybrid	3.5	3	1
ELFIN	9/15/2018	Science	Hybrid	3.6	3	2
CUBERRT	5/21/2018	Tech Demo	Hybrid	10.3	6	1
HaloSat	5/21/2018	Science	Hybrid	12	6	1
RainCube	5/21/2018	Tech Demo	NASA	11.6	6	1
Tempest-D	5/21/2018	Tech Demo	NASA	14	6	5
MarCO	5/5/2018	Tech Demo	NASA	12.6	6	2
MiRaTA	11/18/2017	Tech Demo	Hybrid	4.5	3	1
EcAMSat	11/12/2017	Biology	NASA	10.7	6	1
ISARA	11/12/2017	Tech Demo	NASA	5	3	1
ASTERIA	8/14/2017	Tech Demo	NASA	10.1	6	1
CSUNSat-1	4/18/2017	Tech Demo	Hybrid	2.7	2	1
IceCube	4/18/2017	Tech Demo	NASA	4	3	1
RAVAN	11/11/2016	Tech Demo	NASA	3.9	3	1
MinXSS-1	12/6/2015	Science	Hybrid	3.5	3	1
EDSN	11/4/2015	Tech Demo	NASA	2	1.5	11
GRIFEX	1/31/2015	Tech Demo	Hybrid	4	3	1
RACE	10/28/2014	Tech Demo	NASA	3.8	3	1
SporeSat-1	4/18/2014	Biology	NASA	5.2	3	2
CP8	12/6/2013	Tech Demo	Hybrid	1	1	1
Firefly	11/20/2013	Science	NASA	3.5	3	1
CINEMA-1	9/13/2012	Science	Hybrid	3.1	3	1
CSSWE	9/13/2012	Science	Hybrid	3	3	1
M-Cubed/COVE	10/28/2011	Tech Demo	Hybrid	1	1	1
PSSC-2	7/8/2011	Tech Demo	Hybrid	3.7	2	1
O/OREOS	11/20/2010	Biology	NASA	5.2	3	2
RAX-1	11/20/2010	Science	Hybrid	3	3	2
PharmaSat-1	5/19/2009	Biology	NASA	5	3	2

COMPACT'S KNN ALGORITHM USES PCA

- Intended for early-on, ballpark cost estimates
- Cost estimate is a weighted average of the k most similar missions
- Cost drivers: # U, Mass, Developer Type (NASA/JPL vs. Other)
- Cost drivers are correlated, so we use Principal Components Analysis (PCA) to calculate similarity
- Weighted by the inverse of the distance (i.e. closer missions have more weight)



How does k-Nearest Neighbors (kNN) work?

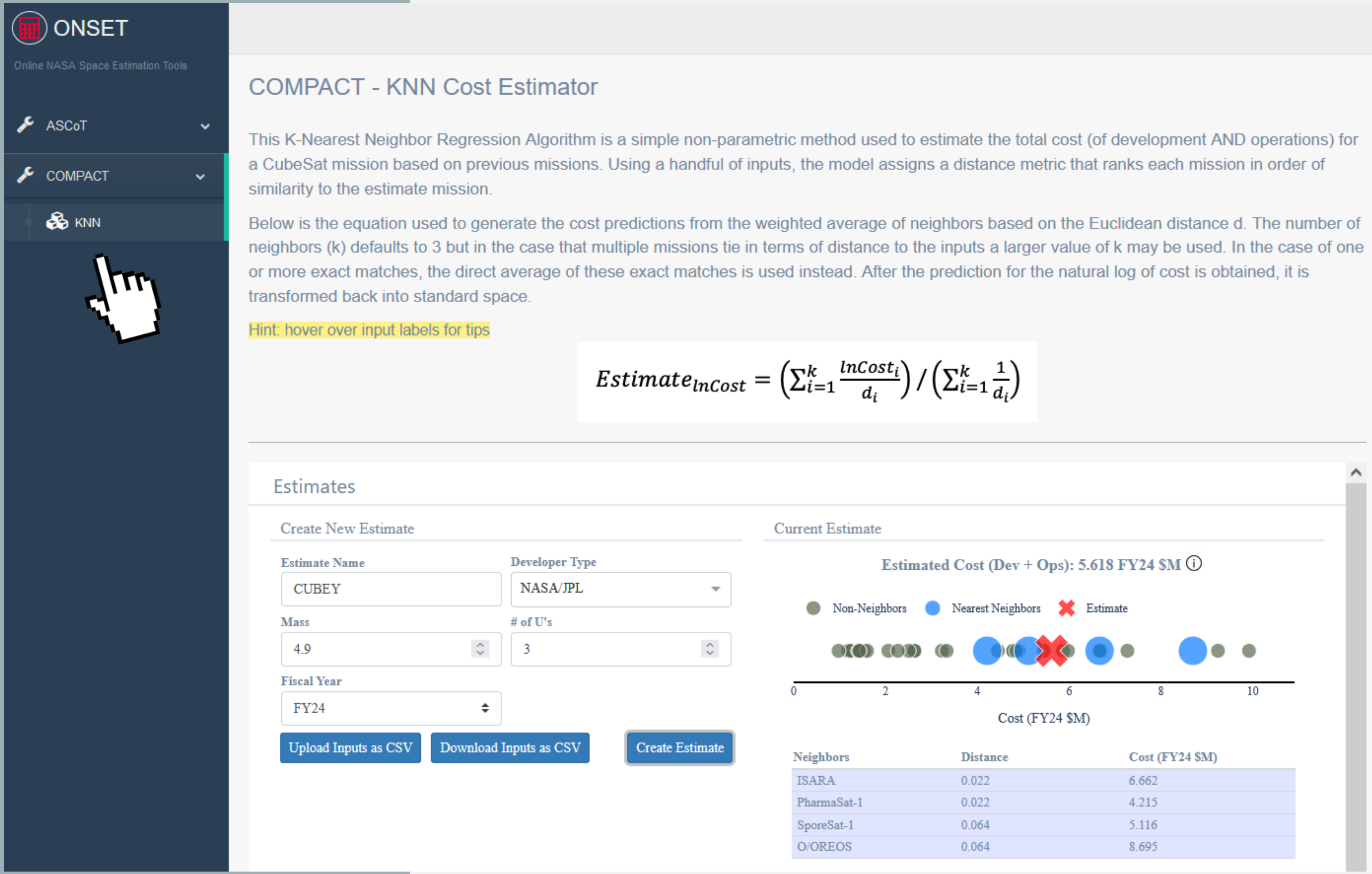


$$\text{Cost(Your Project)} = \frac{\frac{\text{Cost}(P_1)}{d_1} + \frac{\text{Cost}(P_2)}{d_2}}{\frac{1}{d_1} + \frac{1}{d_2}}$$

COMPACT EXAMPLE

You want to propose a new 3U CubeSat to NASA. The CubeSat will be developed at your NASA Center and you estimate that final mass will be 4.9kg.

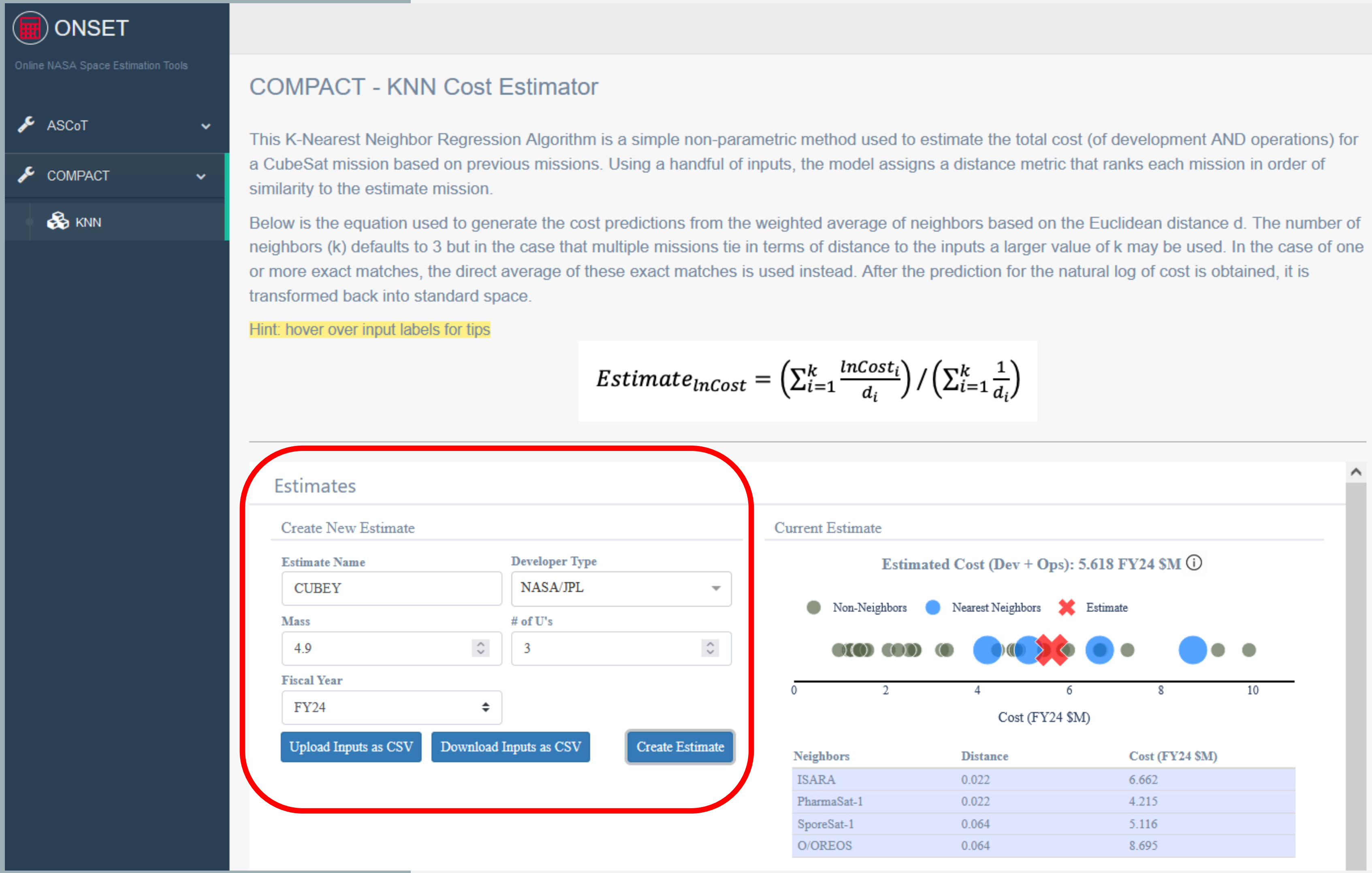
Estimate the total cost of the project.



COMPACT EXAMPLE

You want to propose a new 3U CubeSat to NASA. The CubeSat will be developed at your NASA Center and you estimate that final mass will be 4.9kg.

Estimate the total cost of the project.

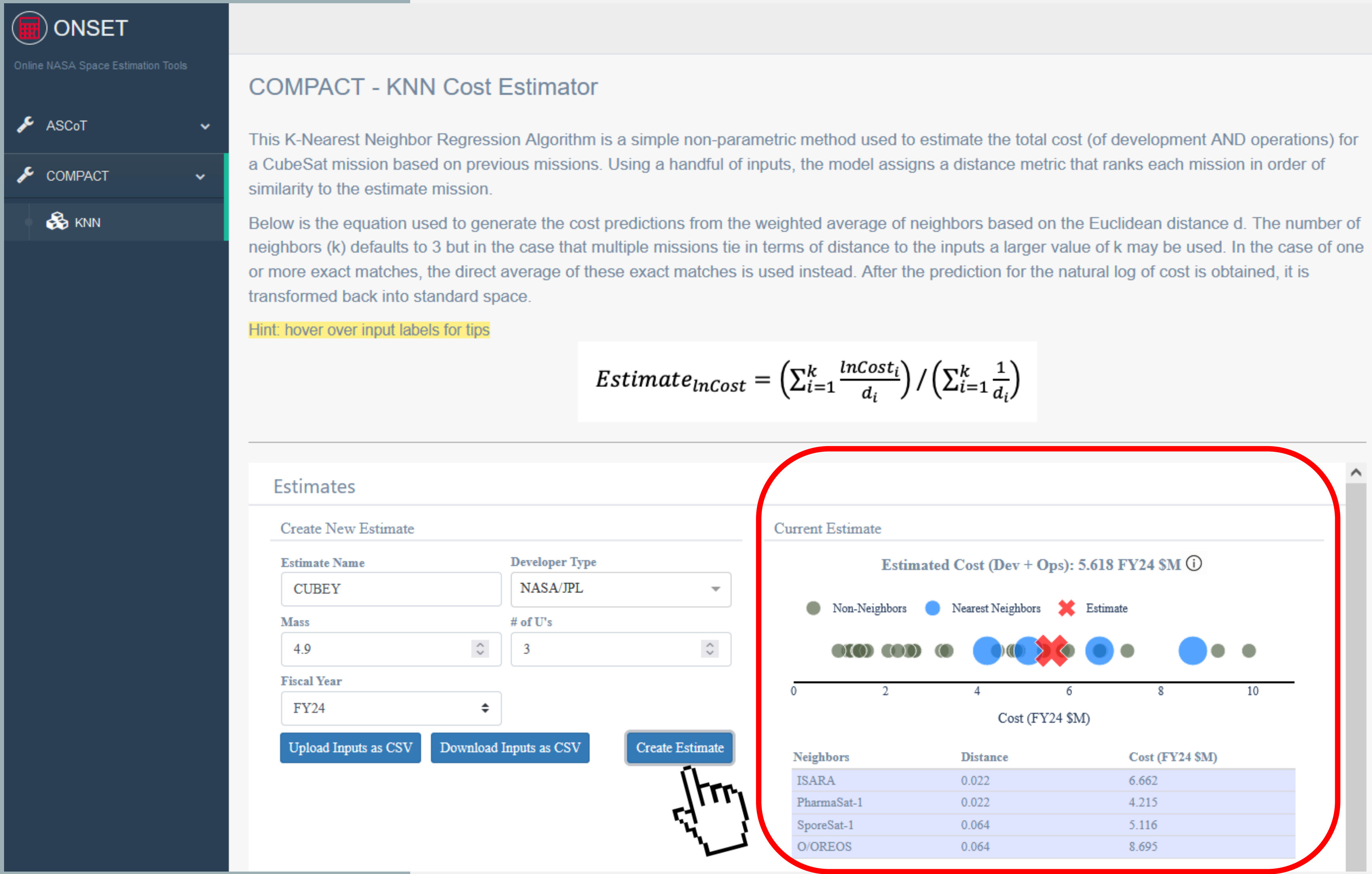


COMPACT EXAMPLE

You want to propose a new 3U CubeSat to NASA. The CubeSat will be developed at your NASA Center and you estimate that final mass will be 4.9kg.

Estimate the total cost of the project.

RESULT: $5.6M FY24

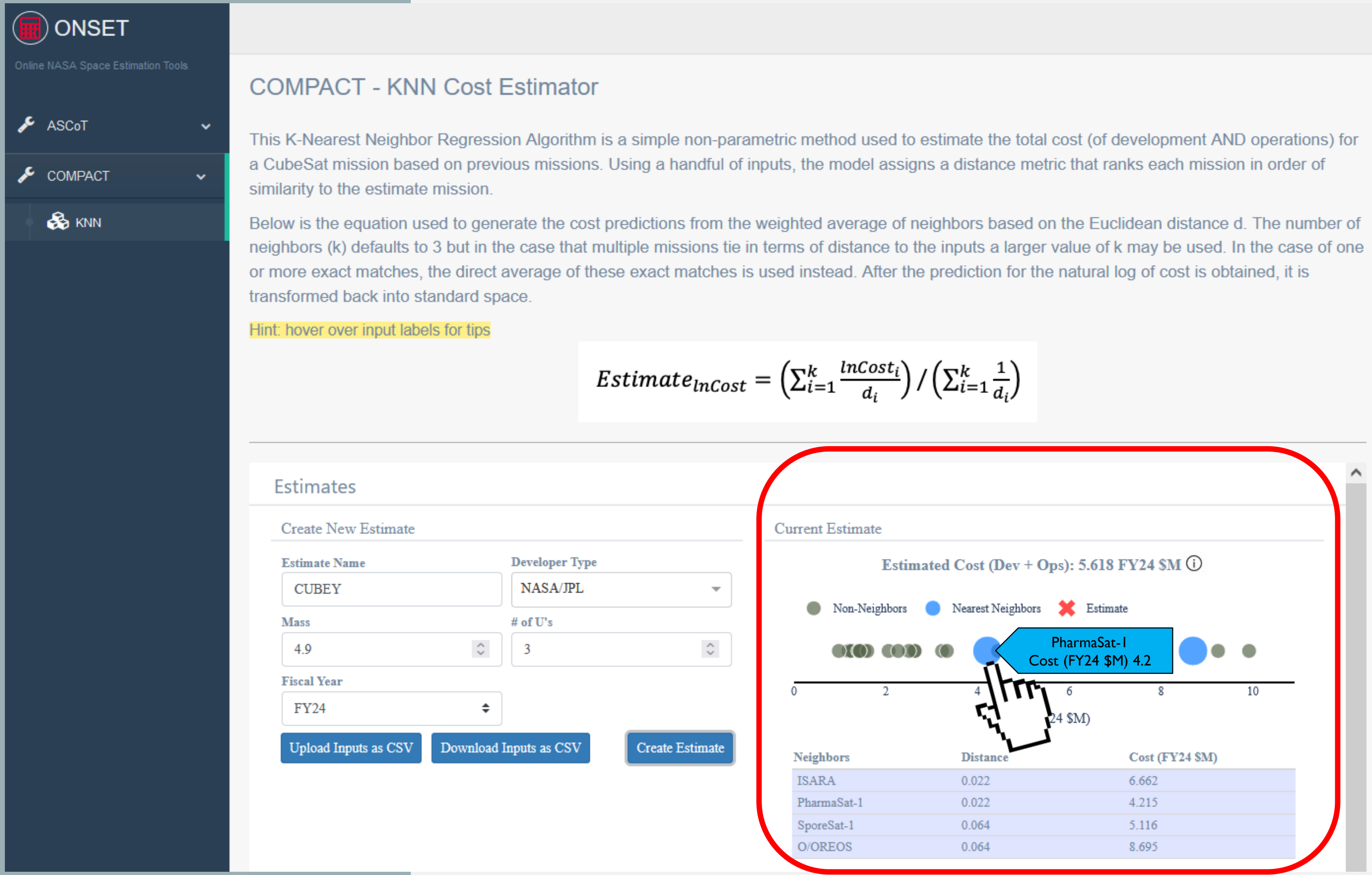


COMPACT EXAMPLE

You want to propose a new 3U CubeSat to NASA. The CubeSat will be developed at your NASA Center and you estimate that final mass will be 4.9kg.

Estimate the total cost of the project.

RESULT: $5.6M FY24



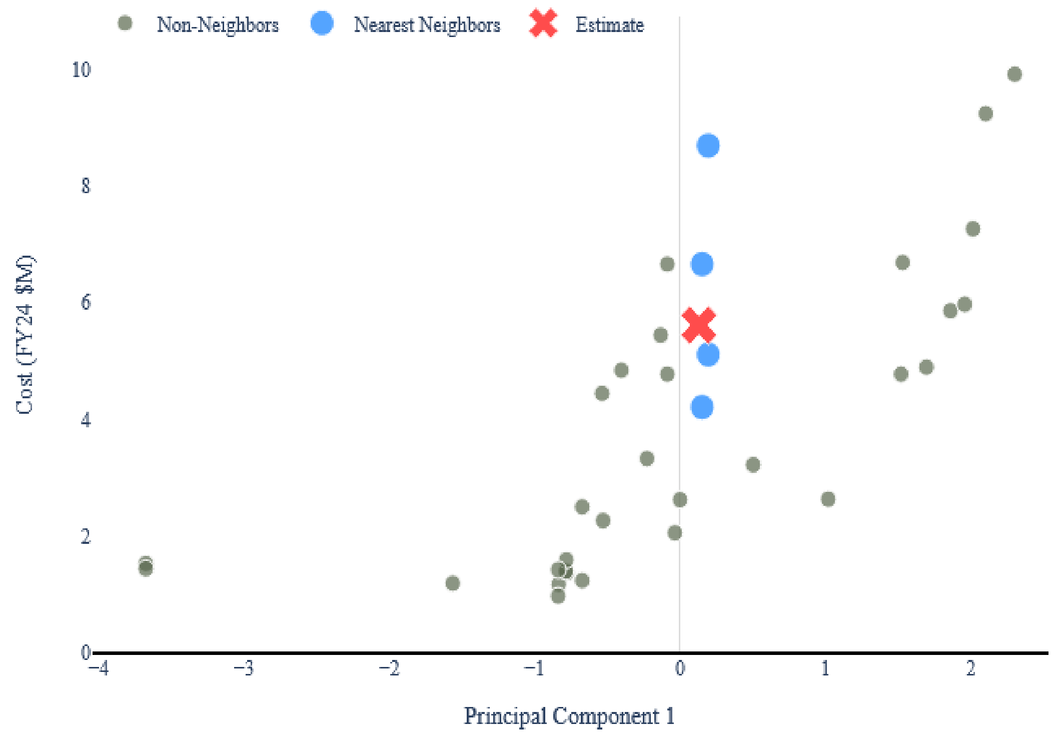
COMPACT EXAMPLE

You want to propose a new 3U CubeSat to NASA. The CubeSat will be developed at your NASA Center and you estimate that final mass will be 4.9kg.

Estimate the total cost of the project.

RESULT: $5.6M FY24

Data



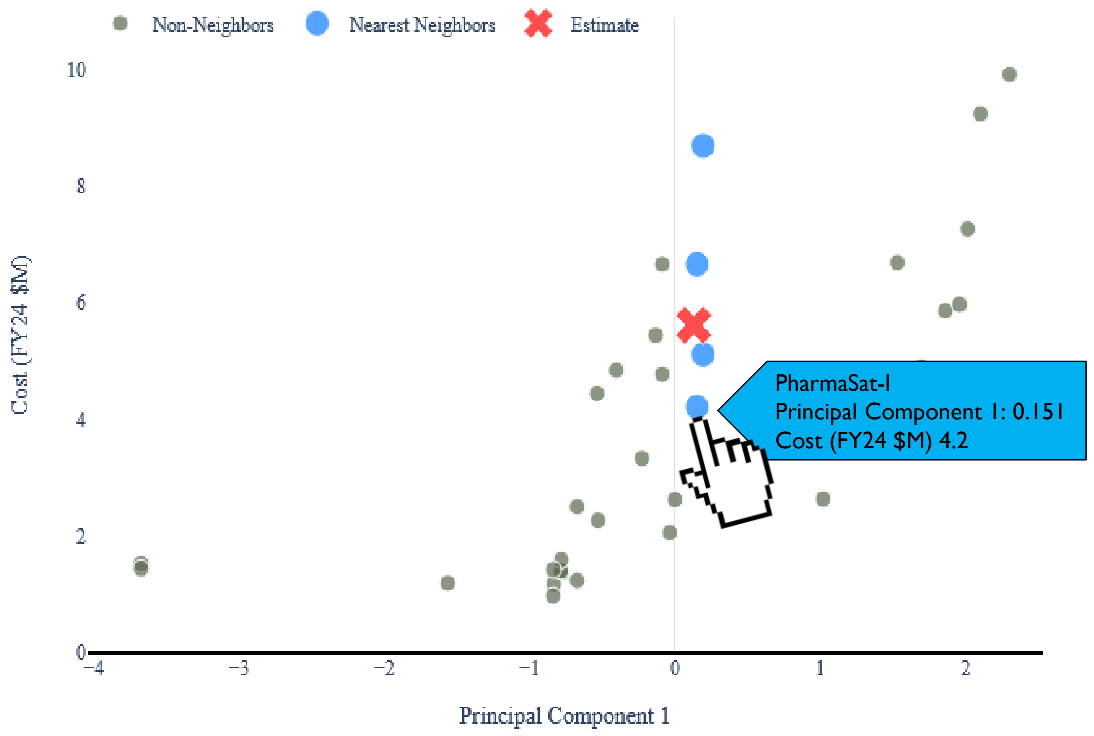
COMPACT EXAMPLE

You want to propose a new 3U CubeSat to NASA. The CubeSat will be developed at your NASA Center and you estimate that final mass will be 4.9kg.

Estimate the total cost of the project.

RESULT: $5.6M FY24

Data



COMPACT CER

Bayesian CER using multi-level modeling.

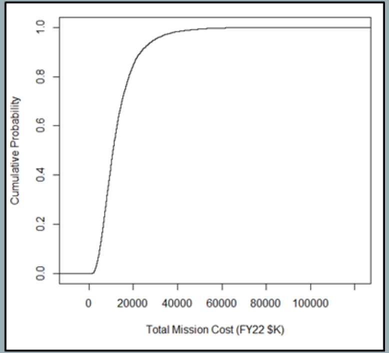
Multiple build estimates using the Crawford model.

Traceable to the COMPACT Database available to the NASA community.

Example Inputs / Outputs

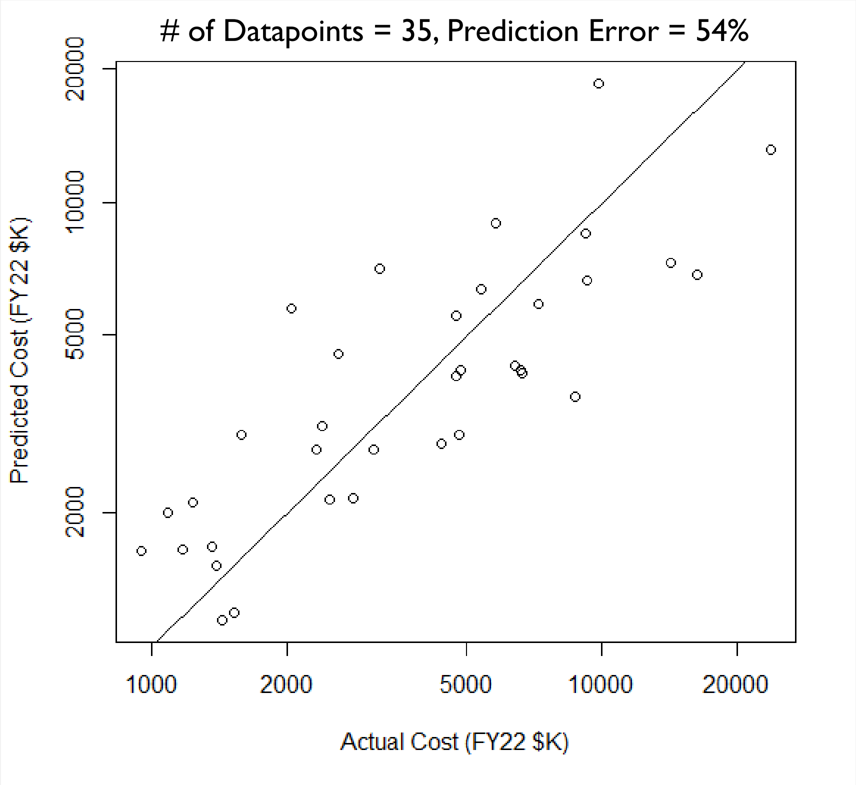
Inputs	Value
NASA or non-NASA	NASA
# of CubeSats	5
# of U's	6
Total Mass (kg)	10
Avg Power (W)	9

FY22 $K	30%	50%	70%
All Units	7,557	10,445	14,574
Unit 1	4,087	5,412	7,242
Unit 2	890	1,466	2,227
Unit 3	714	1,214	1,882
Unit 4	593	1,076	1,720
Unit 5	504	982	1,650



$$Total\ Mission\ Cost\ (FY22\ \$K) = \alpha \times \left(\frac{Mass}{\#\ of\ U's}\right)^{1.41} \times Power^{0.21} \times \underbrace{\left(1 + 0.50 \sum_{n=2}^{\#\ CubeSats} n^{\log_2 0.76}\right)}_{\text{Multiple build term}}$$

$$\text{where } \alpha = \begin{cases} 1808\ if\ NASA/JPL\ developed \\ 1097 otherwise \end{cases}$$



THE FUTURE OF COMPACT

- Continue data collection and normalization
- Continue development of parametric cost models that are accessible to the community
- Continued emphasis on data transparency

THANK YOU

Questions?

Christian Smart (christian.b.smart@jpl.nasa.gov)
Michael DiNicola (Michael.Dinicola@jpl.nasa.gov)