# Automated Spectroscopic Detection And Mapping Using ALMA and Machine Learning Techniques

Andrew Wilkins[1], Steven Cocke[1], Josephine McDaniel[1], John Santerre[1], and Conor Nixon[2]

[1] Master of Science in Data Science, Southern Methodist University, Dallas TX 75275 USA {awilkins,cockes,josephinem,jsanterr}@mail.smu.edu
[2] NASA Goddard Space Flight Center, Greenbelt, MD, USA
conor_a_nixon@nasa.gov
https://science.gsfc.nasa.gov/sed/bio/conor.a.nixon

**Abstract.** In this paper we present a methodology for automating the classification of spectrally resolved observations of multiple emission lines with the Atacama Large Millimeter/submillimeter Array (ALMA). Molecules in planetary atmospheres emit or absorb different wavelengths of light thereby providing a unique signature for each species. ALMA data were taken from interferometric observations of Titan made between UT 2012 July 03 23:22:14 and 2012 July 04 01:06:18 as part of ALMA project 2011.0.00319.S. We first employed a greedy set cover algorithm to identify the most probable molecules that would reproduce the set of frequencies with respective flux greater than $3\sigma$ away from the mean. We then selected a subset of those molecules as present in the atmosphere by specifying a selection threshold and one of two selection metrics. Our model was able to correctly classify 100% of previously discovered molecules in Titan's atmosphere from this data, including Ethyl Cyanide as reported by Cardiner et al. (2015)[2]. One molecule, Formaldehyde, was identified in both selection metrics that was not previously recorded in the atmosphere. The results of our methodology allow for a streamlined approach for molecule classification and anomaly detection in planetary atmospheres.

## 1 Introduction

In 1860, Robert Bunsen (1811-1899) and Gustav Kirchhoff (1824-1887) developed what would come to be known as Kirchhoff's Laws and the foundations of spectroscopy. (2017)[1]. These laws posit that every element and molecule produces a unique spectral line emission pattern. As such, spectroscopy has been and still remains as the principle tool utilized by astrophysicists for determining the composition of planetary and stellar atmospheres. By precisely measuring the frequencies of the absorbed and/or emitted spectral lines in a gas, every element can be distinctly determined.

The primary objective of our work is to automate elemental and molecular classifications contained within planetary atmospheres using spectroscopic data obtained from the Atacama Large Millimeter/submillimeter Array (ALMA), but also to provide a means of detection for unknown molecules. Doing so will automate the preprocessing, analysis, and visual representation of the data, allowing for the most interesting data sets to be quickly identified and selected for full spectral modeling.

The ALMA telescope, located in the Atacama Desert of Chile, is an astronomical interferometer composed of 66 high-precision radio telescopes funded by over seven countries, and costing roughly 1.6 billion USD [8]. The ALMA telescope has the capability of operating in frequency ranges of 30 to 1000 gigahertz (GHz). The data from Titan that we received had already undergone some initial preprocessing, including calibration and deconvoluion ("cleaning"). For a complete description of all transformations performed to preprocess the original raw data, please see subsection 2: Observations of Cordiner et. al. (2015)[2]. The format of the binary raw data returned from the ALMA telescope is outlined in the ALMA Test Interferometer documentation.

Our research builds from the methods described in that paper, and identifies that there is a global need to automate the identification of molecules using spectroscopic data. With millions of known spectral lines available, the task of identifying every type of molecule present can be cumbersome and difficult to do. We present a method for automating this process for spectral observations made using the ALMA telescope array.

## 2 Related Work

Multiple studies have confirmed the presence of molecules such as ethyl cyanide, within Titan's atmosphere [7], using spectra derived from sub-mm and infrared images, as well as emissions found at both poles [6]. It was not until the published paper in 2015 in The Astrophysical Journal Letters titled "Ethyl Cyanide on Titan: Spectroscopic Detection and Mapping Using ALMA" that the first spectroscopic detection was recorded of Ethyl Cyanide. The research performed in this paper is closely related to our own. It described that Ethyl Cyanide was detected in the atmosphere of Titan, and the emission line data that is used comes from the ALMA telescope. However, the method that enabled these discoveries, outside of data preparation and transformation, was manual. In other words, the analysis of the reduced spectra was done by systematic modeling of spectra and exploration of parameter space according to best judgement. There was no automated process for identifying elements based upon their emissions lines.

One group has performed similar work in this area of automation and is discussed in the paper "On The Automated and Objective Detection of Emission Lines in Faint-Object Spectroscopy"[5]. The approach taken was to automate emission line detection in spectral data by calculating signal-plus-noise, and noise only observations via Monte Carlo simulation, and then calculate completeness

and reliability values. Reliability is a measure referred to as the likelihood that a detection is actually correct, and the completeness is a measure referred to as the likelihood that at a given flux, the element is detected. The actual detections are then found by by comparing real data to the Monte Carlo simulations. The data obtained in this study came from a Hectospec spectograph on the MMT Observatory in Mount Hopkins, Arizona, and the dimensions of the data are similar to that produced by the ALMA telescope[1]. Perhaps in future work our methods could be combined.

Another research paper defined methods to verify if a planet truly exists, or if it was masked as something else such as a starspot or plage that emit radial velocity variations [9]. When researchers tried to answer this question, they look for particular known spectral lines emitted that are associated with elements such as calcium, hydrogen & potassium. In this approach, the correlation of spectral lines are calculated against a well-known activity index, so that new activity indices can be identified. The data used in this study was HARPA spectral data, and it also has similar dimensions to our own. The output of this research is a creation of a massive master list of activity-sensitive lines whose fluxes are periodic at the star's rotation period.

## 3 Methodology

### 3.1 Understanding the Data

**Table 1.** Description of the data.

| Data Sources | About This File | Columns |
|---|---|---|
| Win0.clean1.contsub_Jy.rest.scom.c.txt | 3840 Rows, 2 Columns | # Frequency (GHz) |
| Win1.clean1.contsub_Jy.rest.scom.c.txt | | # Flux (Jy) |
| Win2.clean1.contsub_Jy.rest.scom.c.txt | | |
| Win3.clean1.contsub_Jy.rest.scom.c.txt | | |

The initial data set of our project was delivered via four text files that were already preprocessed and include 3840 spectral channels with the following attributes: flux (Jy) recorded out to 9 decimal points, and frequency (GHz) recorded out to 12 decimal points. Table 1 summarizes this information above. There were no missing values or NaN (not a number) values in the input data. The molecules in the atmosphere absorb or emit light at varying wavelengths that are unique to each molecule. The frequency in this case is the independent variable, while the flux received was the dependent variable. Any flux received in

[1] More information may be found at https://iopscience.iop.org/article/10.1086/679285

between two frequency bins was added to the amplitude of the closest frequency bin.

### 3.2 Exploratory Data Analysis

In keeping with traditional best practices when working with an unfamiliar data set, our first objective was to become familiar with it through visualization. Figure 1 is a plot of the data from the four text files all on one axis, while Figure 2 is a reproduction of Cordiner et. al. 2015, Figure 1. From Figure 1, we see that the data is not continuous across the entire spectrum. Each plot on Figure 1 represents just the data available in a given text file. A flux intensity greater than $3\sigma$ (a "spike") is indicated by a dashed gray line. The $3\sigma$ threshold for the flux was used to determine when a flux is significant in keeping with Cordiner et. al. (2015)[2]. In other words, a recorded flux three standard deviations away from the mean flux of that data set would correspond to a potential "signal of interest" from light being reflected by a molecule. This is only a "potential" signal of interest due to that the fact that three standard deviations away from the mean still leaves 0.3% possibility that the signal was actually due to noise. The standard deviation was computed separately for each data set, and minimizes the signal to noise ratio by excluding all flux signals greater than $5\sigma$ in the calculation of the standard deviation.

Fig. 1).

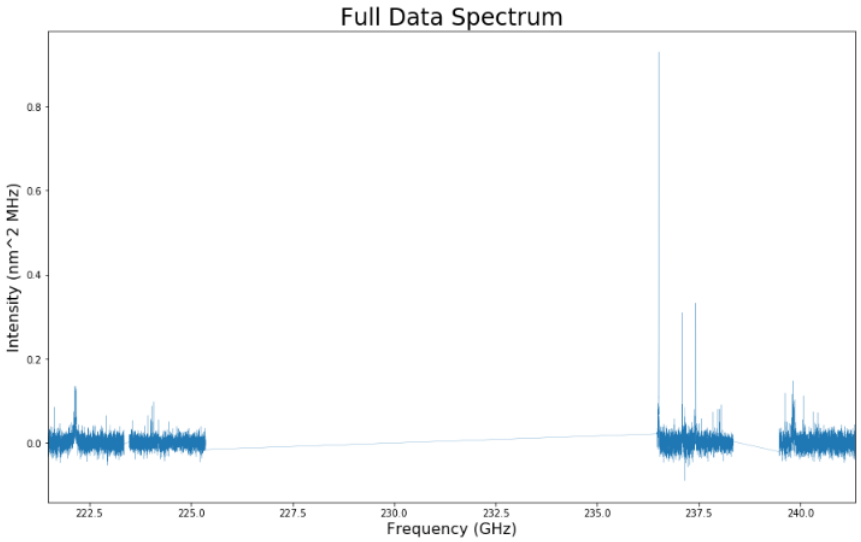



**Fig. 1.** The breaks in the data represent where each text file ends.

Fig. 2).

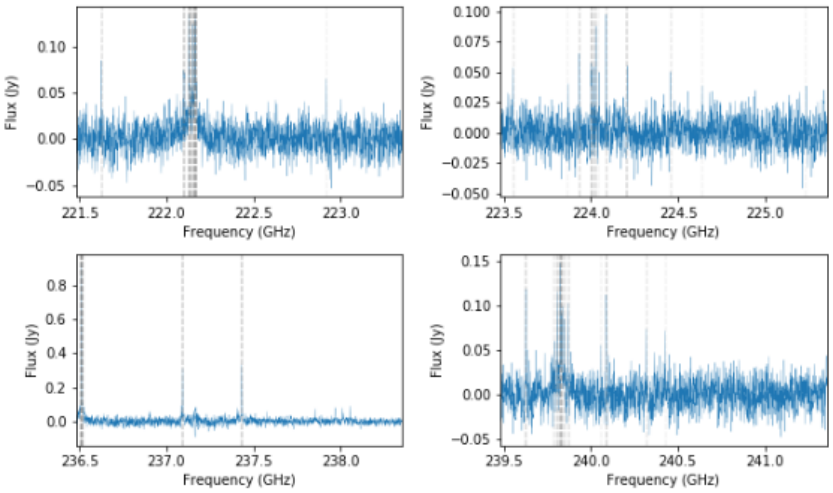



**Fig. 2.** Each vertical gray line represents one frequency bin with greater than $3\sigma$ flux. Darker gray lines indicate regions where several frequency bins in close proximity to on another exceeded the $3\sigma$ threshold.

### 3.3 Obtaining Spectral Emission Data

To determine which molecule corresponds to a spike in flux at a given signal of interest, a comparison must be made to the known spectral lines of every molecule that emits at least one spectral line at each signal of interest within the minimum and maximum frequencies of the input data. For each signal of interest, we query the Splatalogue database using the API available through the `astroquery.splatalogue` python library [4]. Splatalogue, which can be found at https://www.cv.nrao.edu/php/splat/, is a database that allows users to query specific energy and frequency ranges and returns a table object containing information on each molecule that emits at least one known spectral lines within the specified range. Its data is pulled from a combination of 7 different data sources: The Cologne Database for Molecular Spectroscopy (CDMS), Jet Propulsion Laboratory (JPL), National Institute of Standards and Technology (NIST), Toyama Microwave Atlas (ToyaMA), TopModel Lines, Ohio State University (OSU), and Spectral Line Atlas of Interstellar Molecules (SLAIM)[10].

Each query is centered on the signal of interest (in GHz) $\pm$ the channel spacing (488 KHz for the interferometric observations of Titan data). Four additional parameters are passed to the query:

- `Top20 = 'planet'` — Specifies that the molecules of interest come from planetary atmospheres.
- `show_molecule_tag = True` — Ensures the table object also returns the molecule tag.

- `line_lists = ['CDMS', 'JPL']` — Limits the databases searched by the query to CDMS and JPL.
- `line_strengths = 'ls1'` — Specifies that the intensities returned should be pulled from the CDMS and JPL databases.

At present, only the CDMS and JPL databases provide spectral emission data in a format that is usable by our team. Therefore, the other data sources available to Splatalogue will be ignored. The result is a table object containing one row for each result and the following columns:

- Species
- Chemical Name
- Freq-GHz(rest frame,redshifted)
- Freq Err(rest frame,redshifted)
- Meas Freq-GHz(rest frame,redshifted)
- Meas Freq Err(rest frame,redshifted)
- Resolved QNs
- CDMS/JPL Intensity
- S<sub>ij</sub>μ<sup>2</sup> (D<sup>2</sup>)
- S<sub>ij</sub>
- Log<sub>10</sub> (A<sub>ij</sub>)
- Lovas/AST Intensity
- E_L (cm^-1)
- E_L (K)
- E_U (cm^-1)
- E_U (K)
- Molecule<br>Tag
- Linelist

The Chemical Name, Molecule<br>Tag, Linelist, and the frequency that was used to generate the query are then stored for the next phase of data collection.

The Chemical Name value is self-explanatory. The Molecule<br>Tag (Molecule Tag) however, needs some clarification. Splatalogue pulls its information from either the CDMS or JPL databases as specified by the `line_list` parameter. The database it chooses is stored in the Linelist column of the table object. Every molecule on the CDMS and JPL database has its own unique ID. That ID is stored as the Molecule Tag. It should be noted that a molecule found on both the CDMS and JPL databases does not share the same Molecule Tag (ID). Therefore the Linelist *must* be specified along with the Molecule Tag to uniquely identify an entry in the database.

Splatalogue does not provide the entire known spectroscopy for each molecule, but with the parameters specified above, it does provide us with the Molecule Tag and Linelist in order to obtain it. Using this information, we created an API to perform a web-scraping process directly on the originating sources to pull all of the spectral lines for each molecule returned that emits at least one spectral line at any of the identified signals of interest.

The API, named `SpectralQuery`, allows the user to specify a database (currently, CDMS or JPL) and creates a SpectralQuery class object from that input. When an instance of the class is initialized, the URL for the specified database and its associated base URL for results are defined. Using the `BeautifulSoup` python library, the contents of the database are parsed and made ready for queries. At present, the `SpectralQuery` API only contains one method, `getSpectralLines`, which takes the Molecule Tag as a string as input, and returns a `pandas` data frame containing all known spectral line frequencies, uncertainties, and intensities for the specified Molecule Tag. To accomplish this, the method parses through the database contents to find the entry that contains the Molecule Tag in its ID field and then saves the relevant information. The method for parsing through the database is dependent on the database itself however. JPL, for example, provides catalogue files (.cat) that contain the Molecule Tag in the name of the file, thus allowing for an easy search. CDMS on the other hand, provides an .html link for every molecule, but only provides a .cat file for some molecules that are in a format usable for analysis. For this reason, when the `getSpectralLines` method is used on the CDMS database, the .html link is referenced, requiring additional URL parsing to obtain the data we are looking for. Once the spectral emission data is available, regular expression operations are used to format the data and separate entries that might bleed into one another (i.e. formatting 100531.5832-105.9989 into 100531.5832 and -105.9989). Using the `SpectralQuery` API, a python key-value data object (dictionary) is created with each key representing a molecule name identified by Splatalogue, and each value containing the `pandas` data frame returned from `SpectralQuery's getSpectralLines` method using that molecule's Molecule Tag on the database identified by the Linelist. Table 2 illustrates this structure using the molecule Acetone as an example.

**Table 2.** Example dictionary for the molecule Acetone.

| Key | Value | | |
|---|---|---|---|
| | Frequency (MHz) | Uncertainty | Intensity |
| Acetone | 221491.6383 | 0.3403 | -6.6355 |
| | 221499.9092 | 0.0617 | -6.4555 |
| | 221526.9753 | 0.8351 | -6.8137 |

The results are limited to the minimum and maximum frequencies available in the input data. There are in most cases however, many more spectral lines available that fall outside of the range of frequencies in the input data. Figure 3 below is a representation of all spectral emission data for Ethyl Cyanide. The ALMA telescope is only able to capture but a small fraction of the entire spectrum.

Fig. 3).

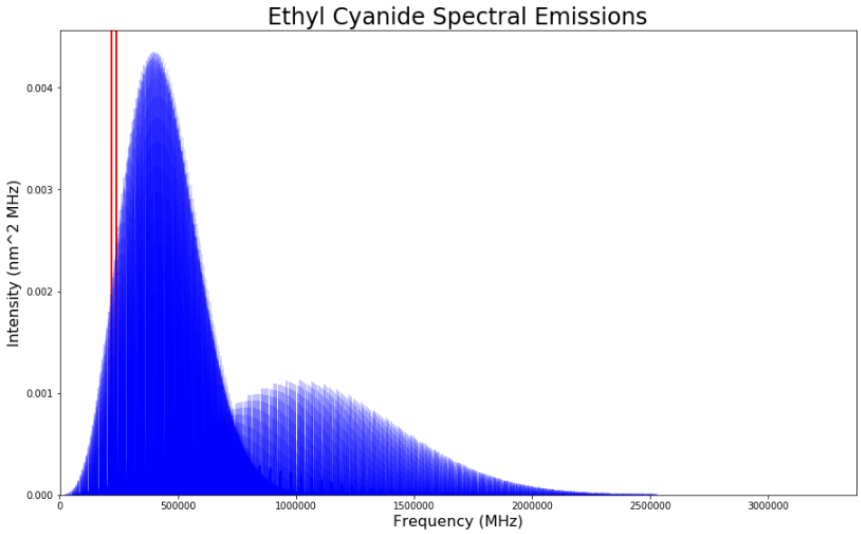



**Fig. 3.** Each vertical blue line represents a known spectral line for Ethyl Cyanide and was retrieved from the CDMS database. The red rectangle represents the 20GHz window for which data is collected from the ALMA telescope.

### 3.4 Combining the Data

The frequencies listed in the dataframe for each molecule were sorted by decreasing intensity and then ranked in that order. Four additional columns were then appended to the dataframes: "Closest," "Flux," "Spike," and "Sigmas." Since the frequencies listed in the CDMS and JPL databases may not align exactly with those in the input data, "Closest" represents the closest frequency from the input data to that listed on CDMS and/or JPL. "Flux" contains the flux of the input data at the "Closest" frequency. If the recorded flux was negative, its value is replaced with 0. The value for "Spike" is True if the flux exceeds the $3\sigma$ threshold of the data set it came from. "Sigmas" are the number of standard deviations away from the mean for the recorded flux. Table 3 provides an example of the structure of the updated dataframe.

For some molecules there were duplicate emission line entries. This could be due to different energy state transitions or recording errors in the database. When duplicate emission line entries are present, only the first entry is retained for the remainder of the analysis.

**Table 3.** Example complete dictionary for the molecule Ethyl Cyanide.

| Key | Value | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | Rank | Frequency (MHz) | Uncertainty | Intensity | Closest | Flux | Spike | Sigmas |
| Ethyl Cyanide | 1 | 237851.8593 | 0.0020 | -2.6292 | 237852.00069 | 0.073258 | True | 3.674600 |
| | 2 | 237405.1700 | 0.0500 | -2.6362 | 237405.19192 | 0.079531 | True | 3.989276 |
| | 3 | 237170.4500 | 0.0500 | -2.6460 | 237170.31212 | 0.084396 | True | 4.233304 |
| | 4 | 223553.6100 | 0.0500 | -2.6706 | 223553.55049 | 0.053545 | True | 4.422450 |
| | 5 | 222918.1747 | 0.0021 | -2.6855 | 222918.23940 | 0.065100 | True | 4.139420 |
| | ... | ... | ... | ... | ... | ... | ... | ... |
| | 146 | 236495.2182 | 0.1404 | -8.5077 | 236495.45997 | 0.051197 | False | 2.568021 |
| | 147 | 221786.4689 | 0.1775 | -8.6681 | 221786.32385 | 0.016719 | False | 1.063085 |
| | 148 | 221665.0508 | 0.1776 | -8.6686 | 221665.22159 | 0.019269 | False | 1.225213 |
| | 149 | 223235.9856 | 0.1157 | -8.6918 | 223236.13285 | -0.006860 | False | 0.000000 |
| | 150 | 223093.5729 | 0.1158 | -8.6923 | 223093.54470 | -0.013184 | False | 0.000000 |

### 3.5 Weighting the Data

As shown in the example table above for Ethyl Cyanide, all of the spectral lines are ordered by descending intensity values. Within a specific frequency range, the data will only comprise of a given number of spectral lines and frequencies. How can one say whether or not Ethyl Cyanide is present in a planetary atmosphere if only a few of the above spectral lines are observed? If the few spectral lines observed are of the highest intensity, we would be more likely to say that the molecule is present than if the few spectral lines observed were of the lowest intensities. This indicates a need for creating a function to weight the spectral lines observed based on the intensity. Equation 1 below defines these weights:

$$W = \begin{cases} \frac{\phi(10^I)}{\sigma} & spike = True \\ 0 & spike = False \end{cases} \tag{1}$$

Where $\phi$ and $I$ are the recorded flux and catalogued intensity, respectively, for a given emission line, and $\sigma$ is the standard deviation of the flux from the data set it came from. The fraction $\frac{\phi}{\sigma}$ has the effect of determining the number of standard deviations away from the mean the given flux is. The only exception to the above equation is for the "Unknown" molecule. Since the intensities for "Unknown" are the same as the flux (by design), the above equation instead takes the form:

$$W = \phi(10^{\phi}) \tag{2}$$

Equation 2 has the effect of causing emissions in the "Unknown" category to carry significantly more weight since the value of the flux is generally much greater than a typical catalogued intensity.

By creating the weights from equation 1 and 2 above, the spectral lines with the greatest intensity and corresponding flux now have a higher weight and therefore carry more importance when determining if a molecule is present or not in an atmosphere. Next, a new `pandas` data frame object was created with each row corresponding to a potential molecule, columns set to the frequencies corresponding to all spikes in the input data, and values equal to the weights calculated above. We refer to this data frame as the weight matrix. The total weight for a given potential molecule then is the sum of the weights in its row of the weight matrix. When all potential molecules are sorted by the their total weight in descending order, we can get a good first order approximation of which molecules are likely to be present in the atmosphere.

### 3.6 Set Cover Method

The next objective was to identify which combination of molecules could reproduce the spike signature observed in the input data. This sort of problem falls within the domain of combinatorial analytics and set cover theory. Vazirani et. al. (2011)[11] provides further guidance into the definitions and methodologies used in set cover theory. To begin, an overview of some useful definitions is in order.

- Universe — The set of all frequencies with fluxes greater than $3\sigma$ will be called the universe. These correspond to the columns of the weight matrix.
- Set — Each possible molecule may have some spikes at a few of the frequencies in the universe. So all the frequencies with a spike will constitute the set for that molecule.
- Weight — Each set also has an associated weight which is a metric of how much of the Universe is being captured by that set.
- Cover — A cover is a collection of sets whose union is the universe. That is, there is a combination of molecules that will reproduce all of the $3\sigma$ spikes that we see in the input data.
- Total Weight — There may be (and will be) multiple covers that can reproduce the universe. Each cover will have an associated total weight which will be the sum of the Weights of the sets that make up the cover.

Our job is to find the most probable covers that can be formed from the fewest number of possible molecules. One approach is to find all possible covers, and then select the smallest cover with the highest total weight. We first tried to accomplish this by implementing a depth first search with the root node as

the set of all possible molecule sets. It was quickly identified however, that it is computationally impossible to find every cover using this method at present. The total number of possible covers from just the molecules with at least one known $3\sigma$ spike in this data set is over $1.18 \times 10^{59}$. We instead focus on finding only the most probable cover by using a greedy approach[11]. We start by defining two empty lists:

- `cover_included` — Contains the names of all included molecules in the cover.
- `cover_freqs` — Contains the set of unique frequencies formed by the union of each set belonging to the molecules included in `cover_included`.

The first step is to identify all molecules which are the sole contributor for a given spike. That is, for spike $A$, the number of all potential molecules with known emission lines at $A$ is exactly 1. This means that every molecule that meets this criteria *must* be included in *any* possible cover because excluding it would result in no feasible covers. We refer to these as "obligatory molecules." The names of each of these molecules are appended to the list of `cover_included` and their sets are appended to `cover_freqs` regardless of their total weight or individual spike weights.

Once all of the obligatory molecules have been included in the cover, we move on to identify the most probable molecules to include. We accomplish this by identifying the row and column of the highest weight in a copy of the weight matrix. The copy is to ensure the integrity of the original weights. Recall, the column corresponds to the frequency of the spike, and the row corresponds to the name of the potential molecule. By selecting the highest weight value, we identify the strongest sources first. If the molecule name isn't already included in `cover_included`, it gets appended to `cover_included` and its set appended to `cover_freqs`. Regardless of whether the molecule with the highest weight gets included or not, that weight gets set equal to 0 in the copy of the weight matrix. Then, the process of finding the next highest weight and testing if the molecule is already included or not is repeated until a viable cover if formed.

Finally, if there is a molecule that is known to not exist in this particular atmosphere, but it has already been included in the cover, the user has the ability to manually remove individual molecules from the cover and generate a new cover excluding those molecules. If the molecule being removed was one of the obligatory molecules, the unique spike it was responsible for gets assigned to the "Unknown" molecule. This step is put in place to ensure that a viable cover will always exist.

### 3.7 Molecule Selection

The greedy set cover approach described in section 3.6 has the advantage of producing a cover that will always include the molecules with the highest weight for each observed spike. However, even if a molecule has a high weight for a given spike, that is not evidence enough that the molecule is in fact present in

the atmosphere. For example, assume molecule $X$ has $n$ emission lines in the range of the input data and carries the greatest weight in the weight matrix for observed spike $A$ at $X_j$, where $X_j$ is not the strongest emission line (say, the $10^{th}$ strongest, for example). If $X$ were present, we would expect to see spikes at each of the $j$ frequencies leading up to $X_j$, or at least a high proportion of those frequencies. If this is not the case, then we must consider that the observed spike be from a different molecule, absorption events in the atmosphere are masking the other spike signatures, or the signal is actually due to noise. We must therefore derive a selection criteria and a threshold at which a molecule is classified as "present" or "not present." Equation 3 defines the selection criteria:

$$F_{molecule} = \sum_{j=1}^{n} \begin{cases} \frac{10^{I_j}}{\sum_{k=1}^{n} 10^{I_k}} & spike_j = True \\ 0 & spike_j = False \end{cases} \tag{3}$$

Where $F_{molecule}$ is the fraction of intensities present for the given molecule, $n$ is the number of emission lines in the range of the input data for a given molecule, $I_j$ and $I_k$ are the CDMS or JPL intensity for the $j^{th}$ and $k^{th}$ entry, and $spike_j$ is the boolean value of whether or not a spike is present in the $j^{th}$ emission line. This selection criteria, we call the "intensity metric," does not account for the value of the flux where a spike is present. Therefore, two entries with identical intensities will carry the same weight regardless of the flux as long as it meets the $3\sigma$ threshold. An alternative to equation 3 that does account for flux is identified here:

$$F_{molecule} = \sum_{j=1}^{n} \sqrt{\left( \frac{10^{I_j}}{\sum_{k=1}^{n} 10^{I_k}} - \frac{\phi_j}{\sum_{k=1}^{n} \phi_k} \right)^2} \tag{4}$$

Where $F_{molecule}$, $n$, $I_j$, $I_k$, $i$ and $j$ are defined as in equation 3, with the addition of the flux of the $j^{th}$ and $k^{th}$ entry, $\phi_j$ and $\phi_k$, respectively. Functionally, it is the square root of the difference of proportions between intensity and flux, squared. In this form, a molecule is selected if $F_{molecule}$ is less than or equal to the specified threshold. This "flux metric" accounts for flux, but ignores whether or not the flux was detected at or above the $3\sigma$ threshold. While this may lead to the detection of weaker signals found below $3\sigma$, it may also introduce molecules that are actually noise. We leave it to the domain experts to decide which metric works best.

## 4 Results

The set cover method identified 34 obligatory molecules and 52 probable molecules in the Titan data. The intensity metric was applied with a threshold of 0.7 and compared against the flux metric with a threshold of 1. The results are listed in table 4.

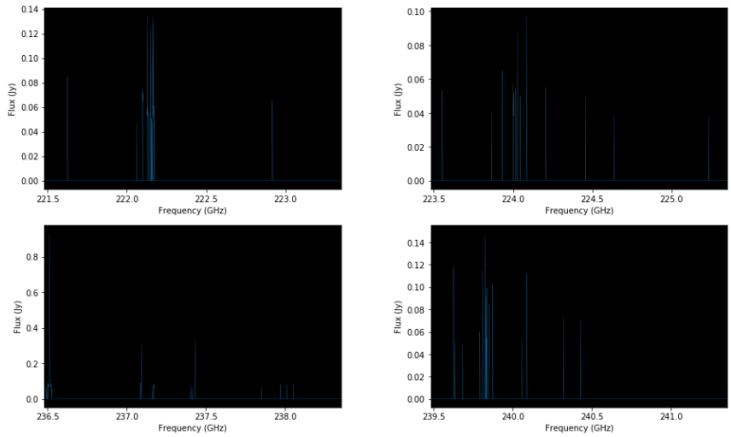
**Table 4.** Selection criteria results.

| Intensity Metric | | Flux Metric | |
|---|---|---|---|
| Molecule Name | Weight | Molecule Name | Weight |
| Unknown | 1.000000 | Cyanoacetylene* | 0.000000 |
| Cyanoacetylene* | 1.000000 | Unknown | 0.253943 |
| Methyl Cyanide* | 0.999961 | Methyl Acetylene* | 0.273384 |
| Ethyl Cyanide* | 0.817057 | Cyanobutadiyne | 0.443561 |
| Methyl Acetylene* | 0.816941 | Formaldehyde | 0.509227 |
| Formaldehyde | 0.745386 | Propadienonethione | 0.649912 |
| | | 1,3-Butadiynyl radical | 0.718551 |
| | | Ethyl Cyanide* | 0.743011 |
| | | Magnesium Cyanide | 0.783229 |
| | | Cyanobutadiynylide anion | 0.803019 |

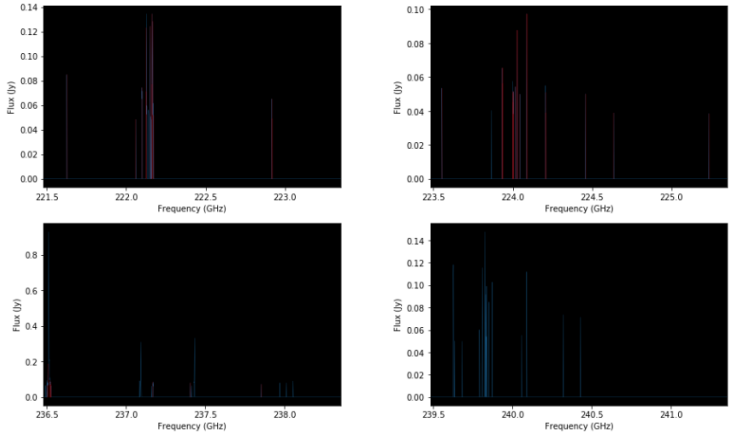
The molecule names appended with an asterisk indicate molecules identified by NASA in [2]. The Unknown molecule is present in both results by design. The intensity metric performed slightly better than the flux metric by correctly identifying all four known molecules in the Titan data. The flux metric was unable to identify Methyl Cyanide despite presenting several other weaker candidates. Interestingly, both metrics identified Formaldehyde as a possible candidate. The formation of Formaldehyde in Earth's atmosphere is a well known process [3]. However, due to its short half-life in air (about one hour), there would have to be a continuous source of for it to be present on Titan. This has piqued NASA's curiosity and is grounds for further research.

Figure 4 shows the spike signatures for the molecules identified in both metrics against all spike signatures. Clearly, there is little difference between 4(b) and 4(c), indicating that the intensity metric may very well be the simpler, better model. It is interesting to note that there are no spikes belonging to any known molecules in the 239.5 - 241.5 GHz range (bottom right plot in 4(b) and 4(c)). This is a curious find and may warrant additional research.

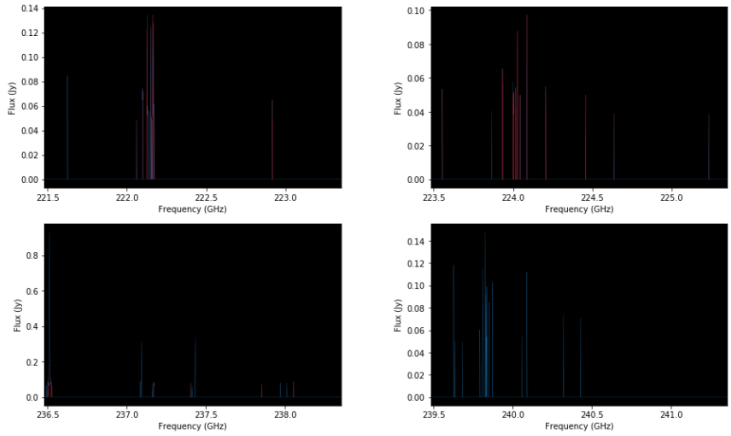
The broader application of our results indicate that automated spectroscopic molecule detection shows promise for a cursory analysis of a planetary atmosphere. A strong case may be made for when both selection metrics identify similar molecules in their output to warrant additional research towards con-




(a)




(b)




(c)

**Fig. 4.** (a) Similar plot as Figure 2, but only those frequencies with associated flux greater than $3\sigma$ are shown. (b) Intensity metric: each identified molecule has their spikes colored in red. (c) Same plot as (b), but for the flux metric.

firming the presence of the identified molecules. In future work, we will test the methods outlined here against other data sets for robustness and accuracy.

## References


1. Carroll, B., Ostlie, D.: An Introduction to Modern Astrophysics. Cambridge University Press (2017)
2. Cordiner, M.A.: Ethyl cyanide on titan: Spectroscopic detection and mapping using alma. The Astrophysical Journal Letters (2015)
3. Debra A. Kaden, Corinne Mandin, G.D.N.P.W.: WHO Guidelines for Indoor Air Quality: Selected Pollutants. World Health Organization (2010)
4. Ginsburg, A., Robitaille, T.: Splatalogue. Astroquery Website
5. Hong, S.: On the automated and objective detection of emission lines in faint-object spectroscopy. Publications of the Astronomical Society of the Pacific **126** (2014)
6. Jennings, D.E., Achterberg, R.K., Cottini, V., Anderson, C.M., Flasar, F.M., Nixon, C.A., Bjoraker, G.L., Kunde, V.G., Carlson, R.C., Guandique, E., Kaelberer, M.S., Tingley, J.S., Albright, S.A., Segura, M.E., de Kok, R., Coustenis, A., Vinatier, S., Bampasidis, G., Teanby, N.A., Calcutt, S.: Evolution of the far-infrared cloud at titan's south pole. Journal of the American Society for Mass Spectrometry (2015)
7. Milligan, D.B., Freeman, C.G., Maclagan, R.G., McEwan, M.J., Wilson, P.F., Anicich, V.G.: Termolecular ion–molecule reactions in titan's atmosphere. Journal of the American Society for Mass Spectrometry (2001)
8. Observatory, E.S.: Alma: In search of our cosmis origins. ESO Website
9. Plavchan, P., Latham, D.: Radial velocity prospects current and future. Exoplanet Program Analysis Group (2015)
10. Splatalogue.net: Splatalogue. https://www.cv.nrao.edu/php/splat/
11. Vazirani, V.V.: Approximation Algorithms. Springer (2011)