

PHYSICAL SENSING AND PHYSICS-BASED MACHINE LEARNING
FOR ACTIONABLE ENVIRONMENTAL INSIGHTS

by

John Waczak

APPROVED BY SUPERVISORY COMMITTEE:

David J. Lary, Chair

Christopher Simmons

David Lumley

Lindsay King

Joseph Izen

Copyright © 2024

John Waczak

All rights reserved

UPDATE REQUIRED!

PHYSICAL SENSING AND PHYSICS-BASED MACHINE LEARNING
FOR ACTIONABLE ENVIRONMENTAL INSIGHTS

by

JOHN WACZAK, BS, PhD

DISSERTATION

Presented to the Faculty of
The University of Texas at Dallas
in Partial Fulfillment
of the Requirements
for the Degree of

DOCTOR OF PHILOSOPHY IN
PHYSICS

THE UNIVERSITY OF TEXAS AT DALLAS

September 2024

ACKNOWLEDGMENTS

UPDATE REQUIRED! Make sure to include lab colleagues *and* ActivePure colleagues and a nod to Dr. Cooper, OSU teachers, and friends...

August 2024

PHYSICAL SENSING AND PHYSICS-BASED MACHINE LEARNING
FOR ACTIONABLE ENVIRONMENTAL INSIGHTS

John Waczak, PhD
The University of Texas at Dallas, 2024

Supervising Professor: David J. Lary, Chair

UPDATE REQUIRED!

TABLE OF CONTENTS

ACKNOWLEDGMENTS	v
ABSTRACT	vi
LIST OF FIGURES	xii
LIST OF TABLES	xiii
CHAPTER 1 INTRODUCTION	1
1.1 Completed Work	3
1.1.1 An Autonomous Robotic Team	3
1.1.2 A Distributed Network of Low-Cost Air Quality Monitors	3
1.1.3	3
1.2 Proposed Work	3
1.3 Timeline	3
CHAPTER 2 BACKGROUND	5
2.1 Global Change	6
2.2 The Role of Sensing	9
2.3 The Role of Computational Modelling	12
2.4 Key Technologies	13
2.4.1 Julia for Scientific Computing	13
2.4.2 Scientific and Physics-based Machine Learning	15
2.5 Water Quality	16
2.5.1 Properties of Aqueous Solutions	16
2.5.2 Reflectance Spectroscopy	16
2.5.3 Solar Geometry	17
2.6 Air Quality	17
2.6.1 Pollution and Particulate Matter	17
2.6.2 Indoor Air Quality	18
2.7 Physics of Chemical Reactions: Chemical Reaction Kinetics	19
2.7.1 Overview	19
2.7.2 Chemical Equilibrium and the Law of Mass Action	20

2.7.3	Reaction Rate Laws	23
2.8	Photolysis	28
2.9	Summary of Chemical Mechanism Kinetics	28
CHAPTER 3 PHYSICAL SENSING		29
3.1	Hyperspectral Imaging	29
3.1.1	Hyperspectral Imaging Sensors	29
3.1.2	Spectral-Spatial Data Collection and Organization	29
3.1.3	Spatial Sampling	30
3.1.4	Spectral Sampling	31
3.1.5	Radiometric Sampling	31
3.1.6	Signal Considerations	32
3.2	Remote Sensing	32
3.2.1	Infrared Sensing Phenomenology	32
3.2.2	Sources of Infrared Radiation	33
3.2.3	Atmospheric Propagation	33
3.2.4	Reflectance and Emissivity Spectra	33
3.3	Coordinated Robotic Teams	34
3.4	In-situ Chemical Sensing in Water	35
3.5	Low Cost Sensors for (Outdoor) Air Quality	35
3.6	The HEART Chamber	35
CHAPTER 4 MACHINE LEARNING METHODS		36
4.1	Adjoint Methods for Optimization	37
4.1.1	Adjoint Methods for Linear Systems	37
4.1.2	Adjoint Methods for Nonlinear Systems	38
4.1.3	Adjoint Methods for ODEs	40
4.2	Exploratory Data Analysis	42
4.2.1	Correlation	42
4.2.2	Mutual Information	42
4.3	Model Training Methodology	42

4.3.1	Data Sampling	42
4.3.2	Model Evaluation	43
4.3.3	Hyperparameter Selection	43
4.3.4	Occam’s Razor and Feature Importance Ranking	43
4.4	Supervised Regression Techniques	44
4.4.1	Neural Networks	44
4.4.2	Gaussian Process Regression	44
4.4.3	Decision Trees	63
4.5	Unsupervised Classification	63
4.5.1	Self Organizing Maps	63
4.5.2	Generative Topographic Maps	68
4.6	Model Ensembling	77
4.6.1	Bagging	77
4.6.2	Boosting	77
4.6.3	Stacking	77
4.7	Uncertainty Quantification via Conformal Prediction	77
4.7.1	Conformalizing Quantile Regression	80
4.7.2	Conformalizing Scalar Uncertainty Estimates	81
4.8	Scientific Machine Learning	82
4.8.1	Physics-Informed Neural Networks	82
4.8.2	Universal Differential Equations	82
4.9	Data Assimilation	82
4.9.1	Kalman Filter	85
4.9.2	Extended Kalman Filter	91
4.9.3	continuous-discrete Extended Kalman Filter	92
4.9.4	3d-Var	95
4.9.5	4d-Var	97
CHAPTER 5 AN AUTONOMOUS ROBOTIC TEAM FOR THE RAPID CHARACTERIZATION OF NOVEL ENVIRONMENTS		100
5.1	Rapid Georectification and Processing of Pushbroom Hyperspectral Imagery	100

5.2	Supervised Regression with Uncertainty Quantification	100
5.3	Methods for Unsupervised Classification of Hyperspectral Scenes in Novel Environments	100
CHAPTER 6 A CHEMICAL DATA ASSIMILATION FRAMEWORK FOR INDOOR AIR QUALITY		101
6.1	Characterization of Photolysis	101
6.2	HEART Chamber Sensing System	101
6.3	Chemical Data Assimilation	101
6.4	An Evaluation of Photocatalytic Ionization	101
CHAPTER 7 A DISTRIBUTED NETWORK OF LOW COST AIR QUALITY SEN- SORS		103
7.1	MINTS Air Quality Network	103
7.1.1	Central Nodes	103
7.1.2	LoRa Nodes	103
7.1.3	MQTT	103
7.2	Real Time Dashboards via Containerization	103
7.3	Making Data Publicly Accessible via S3 and the Open Storage Network . . .	103
CHAPTER 8 TIME SERIES METHODS FOR AIR QUALITY		105
8.1	Time-Series Methods for Uncertainty Quantification	105
8.1.1	Metrics for Representative Uncertainty of combined Pseudo-Observations	105
8.1.2	Uncertainty Quantification with Temporal Variograms	105
8.2	Physics Informed modeling techniques for Air Quality Data	105
8.2.1	Embedding Theorems	105
8.2.2	Koopman Theory	105
8.2.3	HAVOK + SciML = Nonlinear Modeling with External Forcing . . .	105
8.2.4	Hamiltonian Neural Networks	105
CHAPTER 9 LIMITATIONS AND FUTURE WORK		109
9.1	Robot Team	109
9.1.1	Super Resolution	109

CHAPTER 10 CONCLUSIONS	110
APPENDIX A CHEMICAL REACTION MECHANISMS	111
A.1 Simple Ion Mechanism	111
A.1.1 Bimolecular Reactions	111
A.1.2 Trimolecular Reactions	166
A.1.3 Photolysis Reactions	186
A.2 Master Chemical Mechanism	194
REFERENCES	195
BIOGRAPHICAL SKETCH	199
CURRICULUM VITAE	

LIST OF FIGURES

2.1	An illustration of the broad range of reaction time scales from the long-lived nuclear decay to rapid degradation of molecules by photolysis. Figure taken from (Arnaut and Burrows, 2006)	20
4.1	A standard linear regression fit on some noisy data.	46
4.2	An example of polynomial regression for which we fit a quadratic polynomial to some noisy data.	51
4.3	A Gaussian Process fit to the training data illustrating the predication (means) and a $\pm 2\sigma$ uncertainty interval. We can clearly see how the uncertainty is larger for inputs far away from supplied training data.	61
4.4	Left: the original Gaussian Process obtained with our default choice of kernel parameters. Right: A much better Gaussian Process obtained after performing hyper-parameter optimization on the kernel parameters. Note how the mean function still does an excellent job fitting the data points while <i>also</i> minimizing the uncertainty of the fit.	64
4.5	A Self Organizing Map with nodes configured in a rectangular grid topology with weight vectors randomly initialized.	66
4.6	A trained Self Organizing Map of 25×25 cells in a hexagonal topology trained on a dataset consisting of the three colors red, green, and blue.	69
4.7	An example of conformal prediction for a model estimating a noisy one-dimensional target. The blue shaded region illustrates the learned confidence interval. Image taken from (Altmeyer, 2023)	79

LIST OF TABLES

CHAPTER 1

INTRODUCTION

The rapid pace of global change poses a significant and everpresent threat to human prosperity. To facilitate the development of remediation technologies and enable effective mitigation strategies, we must make *data-driven decisions*. However, the limitations posed by the lack of highly available, highly resolved data coupled together with the computational challenges posed by direct simulation of physics *at scale* severely constrains our ability to make low uncertainty predictions to meaningfully address these challenges. The goal of this dissertation is to expand the boundary of what is possible in the domain of physical sensing by leveraging the ever-expanding field of machine learning in a *principled way* to produce actionable insights.

we should not throw the baby out with the bath water. Nor should we attempt to reinvent the wheel... we can use machine learning to *fill in the gaps* where our current theory can not easily provide a direct link between the data we have and the target we wish to model but a relationship is easily justified on physical grounds. Further, we can use our knowledge of physical laws to impose strong constraints on the space of available models that may significantly improve the capability of machine learning models to extrapolate beyond the boundaries of their training data sets.

make the point that for physical theories are typically only amenable to analytic techniques when sufficiently constrained to domains with high symmetry or where nonlinear interactions can be controlled so that linear effects dominate over bounded of higher order effects (re: perturbation theory). For common real world scenarios where edges are sharp, functions aren't smooth, and approximating assumptions can not be readily applied, we typically must resort to numerical methods which seek to solve the dynamical laws on computational domains with well posed boundary and initial conditions.

In the first proposed project, we will apply machine learning techniques to provide a sensor calibration where theory suggests

To that end, this work focuses on improving sensing capabilities in three key domains: water quality, outdoor air quality, and indoor air quality.

This goal is pursued by applying physics informed approaches together with a suite of sensing and computational technologies, implementing the reusable paradigm of software defined sensors, i.e. physical sensing elements wrapped in a software layer. This software layer can serve a variety of purposes such as calibration and the provision of enhanced or derived data products. It is part of a broader effort in the MINTS-AI laboratory at the University of Texas at Dallas. Where MINTS-AI is an acronym, Multi-Scale Multi-Use Integrated Intelligent Interactive Sensing in Service of Society for Actionable Insights.

Comprehensive environmental sensing is a timely and beneficial endeavor for a variety of reasons. The growing awareness of major environmental issues such as climate change, pollution, and habitat loss necessitates effective environmental monitoring and management. Comprehensive environmental sensing can provide real-time data on air and water quality, weather patterns, and other environmental factors, assisting in the identification and resolution of environmental issues. This assists in the development and implementation of policies and strategies aimed at reducing environmental impact and increasing sustainability. Given that, for instance, air quality can have significant effects on human health, this has particular societal value.

Comprehensive sensing of the environment can improve decision-making. The real-time and accurate data provided by environmental sensors can aid in informed decision-making regarding various aspects such as traffic management, industrial regulation, and crop planning. For instance, data on air quality can be used to inform decisions about reducing pollution levels, while data on weather patterns can help farmers to plan their crops and reduce water usage. Comprehensive sensing of the environment can be instrumental in emergency response. Real-time data on weather patterns, air quality, water levels and resources,

and seismic activity can help emergency responders to prepare for and respond to natural disasters such as hurricanes, floods, and earthquakes. The quick and accurate information can enable effective and timely response, potentially saving lives and reducing the impact of the disaster.

Many advances in technology have enabled the creation of comprehensive sensing systems that can monitor and analyze data from various sensors and devices in real-time. In this thesis we use a range of technologies including autonomous robotic teams [Dunbabin2012; Rubenstein2014; Chen2017], hyperspectral imaging [Plaza2009; Li2018; Zhu2017], mesh networks utilizing the Internet of Things (IoT) [Gubbi2013; Atzori2010; Al-Fuqaha2015], machine learning (ML) [Goodfellow2016; LeCun2015; Jordan2015], edge computing, high-performance computing, wearable sensors and modern high-performance dynamic programming languages such as Julia [Bezanson2017] designed for numerical and scientific computing. These technologies have facilitated the collection and processing of large amounts of data from multiple sources, resulting in more accurate and comprehensive environmental monitoring.

Today, the most well known application of machine learning techniques

1.1 Completed Work

1.1.1 An Autonomous Robotic Team

1.1.2 A Distributed Network of Low-Cost Air Quality Monitors

1.1.3

1.2 Proposed Work

1.3 Timeline

NOTE: we may want to merge the Introduction with the Physical Context chapter.

CHAPTER 2

BACKGROUND

The impacts of global change are everpresent. Forest fires, hurricanes, and other extreme weather events...

In this “global” scale introduction, we should introduce the context of global change and the need for improved sensing and modeling to provide actionable insights at a pace that meets societal/human needs. We can then discuss how big data + machine learning can help fill in the gaps where our theoretical knowledge is incomplete, expensive (money and computational) to simulate directly, or plagued by initial condition sensitivity (e.g. Chaos).

Finish with a transition paragraph providing an overview of each of the following chapters
For Water Quality:

- chemical quantification (crude oil, algal blooms, terrorist events, etc...)
- chemical identification (can we identify new constituents in the water?)

Outdoor Air Quality:

- How can we model the uncertainty of low cost sensors for real-time applications. This will have applications on real-time decision making, calibration, QA/QC, etc...
- Can we leverage data to effectively model the dynamics of local air quality without the need for complicated microphysics simulations? (We can say something here about the need to go beyond thermodynamic equilibrium)
- Can we learned models provide new insights into real physics? E.g. what do the learned terms of our SciML extended HAVOK model tell us? How can we interpret the forcing function? What do the learned coordinates of the HNN represent? How can we interpret dynamics on the Hamiltonian surface?

Indoor Air Quality:

- Can we leverage cutting edge measurement techniques to effectively model indoor chemical kinetics?
- What is the role of ion chemistry in indoor air quality?
- By combining observed species concentrations with highly detailed kinetics, can we infer the presence and role of species below detectable limits?

For the proposal, let's end the introduction with a timeline (we can use Dr. Lary's Gantt chart).

2.1 Global Change

Global change refers to the significant and long-term alterations in the Earth's physical, chemical, and biological systems, resulting from natural and human-induced processes (Edenhofer et al., 2014; Masson-Delmotte et al., 2018; United Nations, 2015). This includes changes in the climate, land use, biodiversity, and biogeochemical cycles, as well as interactions among these systems. Global change can have profound impacts on natural and human systems, including altered weather patterns, sea level rise, increased frequency and severity of extreme events, loss of biodiversity and ecosystem services, and effects on human health and well-being. Understanding and managing global change is a critical challenge facing society today, requiring interdisciplinary approaches and collaboration across sectors and regions.

Global change can have a range of impacts on society, including environmental, social, and economic effects. Some of the aspects of global change that have the biggest impact on society include:

- Climate Change: Climate change, driven by human activities such as burning fossil fuels, deforestation, and land-use changes, has impacts on natural systems such as

ocean acidification, sea level rise, and changes in precipitation patterns. These impacts can have cascading effects on human systems, including impacts on food security, water availability, and health.

- **Biodiversity Loss:** Global change can lead to the loss of biodiversity, which can have impacts on ecosystem functioning and services, such as pollination, pest control, and carbon storage. These impacts can have indirect effects on human well-being, including impacts on food security, health, and cultural heritage.
- **Land Use Change:** Land use change, such as deforestation, urbanization, and agriculture, can have impacts on natural systems such as soil quality, water availability, and biodiversity. These impacts can have direct and indirect effects on human systems, including impacts on food security, water availability, and cultural heritage.
- **Economic and Social Inequality:** Global change can exacerbate economic and social inequality, with impacts on access to resources, health, and well-being. These impacts can have cascading effects on the ability of societies to adapt and respond to global change.
- **Human Health:** Global change can have significant impacts on human health (World Health Organization, 2018; Costello et al., 2009; Haines et al., 2006), both directly and indirectly, for example:
 - **Heat-related Illness:** As temperatures increase due to global warming, there is an increased risk of heat-related illness, including heat exhaustion and heat stroke, particularly in vulnerable populations such as the elderly, young children, and outdoor workers.
 - **Air Pollution:** Global change can lead to increased air pollution, including from sources such as wildfires and fossil fuel combustion. Exposure to air pollution

can increase the risk of respiratory and cardiovascular diseases, including asthma, chronic obstructive pulmonary disease (COPD), and heart disease.

- Vector-borne Diseases: Changes in temperature and precipitation patterns can affect the distribution and abundance of disease vectors such as mosquitoes and ticks, leading to increased risks of vector-borne diseases such as dengue fever, malaria, and Lyme disease.
- Waterborne Diseases: Changes in precipitation patterns and water quality can increase the risk of waterborne diseases, including cholera and other diarrheal diseases.
- Food Security: Global change can affect food production and availability, leading to food shortages and malnutrition, particularly in vulnerable populations such as children and pregnant women.

Effectively addressing these aspects of global change requires interdisciplinary approaches and collaboration across sectors and regions, as well as a commitment to sustainable development and equitable solutions. Adaptation and mitigation are two strategies for addressing global change, which differ in their focus and approach.

Adaptation involves taking measures to adjust and respond to the impacts of global change that are already occurring or are expected to occur in the future. This can include actions such as building sea walls to protect against sea level rise, developing drought-resistant crops, and improving public health infrastructure to address the increased risk of vector-borne diseases. Adaptation strategies aim to reduce the vulnerability of human and natural systems to the impacts of global change and increase their resilience.

Mitigation involves taking measures to reduce the drivers of global change, such as greenhouse gas emissions, land use change, and deforestation. This can include actions such as increasing energy efficiency, shifting to renewable energy sources, and reducing waste and

consumption. Mitigation strategies aim to address the root causes of global change and reduce its severity and impact.

Both adaptation and mitigation are important strategies for addressing global change, but they differ in their focus and approach. Adaptation strategies focus on responding to the impacts of global change that are already occurring or are expected to occur in the future, while mitigation strategies focus on reducing the drivers of global change and preventing its impacts from occurring in the first place. A comprehensive approach to global change will require both adaptation and mitigation strategies, as well as efforts to promote sustainable development and equitable solutions.

2.2 The Role of Sensing

Sensing technologies can play a critical role in both adaptation and mitigation efforts by providing data and information that can inform decision-making and improve the effectiveness of strategies (United Nations Environment Programme, 2017; National Research Council, 2010; Centre for Ecology and Hydrology, 2017).

In adaptation efforts, sensing technologies can provide real-time data on environmental conditions such as temperature, precipitation, sea level, air quality, as well as on the status and health of ecosystems and wildlife. This information can be used to inform early warning systems for natural disasters, to track the spread of vector-borne diseases, and to monitor the impacts of climate change on biodiversity and ecosystem services. Sensing technologies can also provide data on the effectiveness of adaptation measures, such as the performance of sea walls and other infrastructure.

In mitigation efforts, sensing technologies can provide data on greenhouse gas emissions and other drivers of global change, as well as on the effectiveness of mitigation measures such as renewable energy and carbon capture and storage. Sensing technologies can also be

used to monitor and manage land use changes such as deforestation and urbanization, and to track the impacts of these changes on ecosystems and carbon storage.

Overall, sensing technologies can provide critical data and information for both adaptation and mitigation efforts, helping to improve decision-making and increase the effectiveness of strategies. The integration of sensing technologies with other tools such as modeling and data analysis can also help to identify new strategies and solutions for addressing global change. There are various sensing technologies and approaches used for monitoring the global environment. Here are some of the key ones:

- Remote Sensing: This technology involves using satellites and other airborne platforms to collect data on the Earth’s atmosphere, land, and oceans. Remote sensing provides information on environmental parameters such as temperature, humidity, air quality, land use and land cover, and ocean temperature, salinity, and sea level (Thenkabail, 2019; Buyantuyev and Wu, 2017; Gamon et al., 2016; Wang et al., 2017; Pasher et al., 2019). Some examples of remote sensing include:
 - Lidar: This technology uses laser pulses to measure distance and can be used to create detailed three-dimensional maps of the environment. Lidar is commonly used to measure forest canopy height, but can also be used to measure atmospheric conditions such as cloud cover and aerosol concentrations.
 - Imaging Spectroscopy: This technology uses a combination of imaging and spectroscopy to measure the reflectance of different wavelengths of light. Imaging spectroscopy can be used to identify and map different types of vegetation and minerals, and can provide information on the health of plant communities.
 - Unmanned Aerial Vehicles (UAVs): These are remote-controlled or autonomous aircraft that can be equipped with sensors for remote and in-situ environmental monitoring. UAVs can be used for mapping and monitoring of large areas, and can collect high-resolution data on environmental conditions.

- In-Situ Sensors: These sensors are used to collect data directly from the environment at the location of interest. They can measure environmental parameters such as temperature, pressure, and humidity, as well as water quality and soil moisture. In situ sensors are commonly used in marine environments to measure ocean temperature, salinity, and other properties. Some examples of in-situ sensing include:
 - Weather Stations: These are automated weather monitoring systems that collect data on atmospheric conditions such as temperature, humidity, barometric pressure, wind speed and direction, and precipitation. Weather stations can be installed on land or in the ocean to provide continuous monitoring of environmental conditions.
 - Ground-Based Sensors: These sensors are used to monitor the quality of air, water, and soil. They can detect and measure pollutants such as carbon dioxide, nitrogen dioxide, ozone, sulfur dioxide, and particulate matter. Ground-based sensors are installed in various locations such as cities, industrial sites, and rural areas to provide localized environmental monitoring.
 - Acoustic Sensors: These sensors are used to monitor environmental noise levels, including noise from traffic, industrial sources, and natural sources such as wind and waves. Acoustic sensors can provide information on noise levels over time and across different locations.

Overall, these sensing technologies play a critical role in monitoring the global environment and can provide valuable information for environmental research, management, and policy-making.

2.3 The Role of Computational Modelling

Computer modeling can play a valuable role in both understanding and predicting global change (Chen et al., 2019; Hantson et al., 2016; DeLucia et al., 2021; Oleson et al., 2013; Clark et al., 2016). For example:

- **Climate Modeling:** Computer models can be used to simulate the Earth’s climate system and predict future climate conditions. These models can incorporate data on greenhouse gas emissions, land use changes, and other factors to project how the Earth’s climate will change over time.
- **Ecosystem Modeling:** Computer models can be used to simulate how ecosystems will respond to changes in environmental conditions, such as changes in temperature, precipitation, and atmospheric composition. These models can help predict how changes in ecosystems will impact biodiversity, ecosystem services, and human well-being.
- **Carbon Cycle Modeling:** Computer models can be used to simulate the global carbon cycle, which is the exchange of carbon between the Earth’s atmosphere, land, and oceans. These models can help predict how changes in carbon emissions and land use will impact atmospheric carbon dioxide concentrations and global climate.
- **Air Quality Modeling:** Computer models can be used to simulate air quality, including the dispersion of pollutants in the atmosphere. These models can help predict how changes in emissions and atmospheric conditions will impact air quality and human health.
- **Hydrological Modeling:** Computer models can be used to simulate the movement of water through the Earth’s hydrological cycle. These models can help predict how changes in precipitation, land use, and other factors will impact water availability, quality, and distribution.

Overall, computer modeling can provide valuable insights into the complex processes and interactions that drive global change. These insights can inform policy decisions and help guide efforts to mitigate and adapt to the impacts of global change.

2.4 Key Technologies

2.4.1 Julia for Scientific Computing

Julia is designed to combine the ease of use and high-level abstractions of languages like Python with the performance of compiled languages like C++, achieving a unique combination of speed and productivity for numerical and scientific computing. Julia is a high-level, high-performance programming language designed for numerical and scientific computing. It combines the ease of use and readability of Python with the speed and efficiency of Fortran or C. Julia has a wide array of scientific computing functionality, making it a powerful language for numerical analysis, data science, and engineering. It has built-in support for arrays and linear algebra, as well as packages for differential equations, optimization, probabilistic programming, data analysis and visualization, parallel and distributed computing, and machine learning. Julia's combination of performance, expressiveness, and flexibility make it an excellent choice for scientific and engineering applications, allowing for high-level abstractions and rapid prototyping, while still providing low-level control and efficient execution.

Here are some examples of what can be done easily in Julia that may not be as easy or efficient in other widely used scientific computing languages such as Python or Fortran:

- **Multiple dispatch:** Julia has a powerful multiple dispatch system that allows for generic programming and efficient function overloading. This allows for more flexible and expressive code compared to traditional object-oriented programming (OOP) in Python. Multiple dispatch allows a function to behave differently based on the types and/or number of arguments passed to it. In other words, the behavior of a function can be dispatched based on the specific types and/or number of arguments passed to it.

- Just-in-time (JIT) compilation: Julia's JIT compiler translates high-level Julia code into optimized machine code, making Julia programs run nearly as fast as C or Fortran. In contrast, Python code is interpreted, and Fortran requires pre-compilation.
- Distributed computing: Julia has built-in support for distributed computing, making it easy to parallelize and scale up computations across multiple processors or machines. This is not as easy to do in Python or Fortran.
- Units and Error Propagation: The Units package in Julia provides a powerful and flexible framework for handling physical units in computations, useful for error propagation and dimensional analysis, helping to ensure that the results are accurate, consistent, easy to interpret, and dimensionally consistent.
- Built-in unit testing: Julia has a built-in testing framework that makes it easy to write and run unit tests for your code, ensuring that it works correctly.
- ISO characters: Julia supports the use of Greek and other ISO characters in variable and function names, which can make code more readable and expressive, especially in mathematical or scientific contexts.
- Interactive data visualization: Julia has a number of powerful data visualization packages, such as Plots.jl and Makie.jl, that allow for interactive, high-performance data visualization.
- Package management: Julia has a sophisticated package manager that makes it easy to install, manage, and use third-party packages in your code. This is not as easy to do in Fortran, and while Python has a package manager, Julia's package manager is faster and more reliable.

- Inline C/Fortran/Python/R/Matlab code: Julia allows for inline C, Fortran, Python, R or Matlab code, making it easy to use existing libraries and code written in these languages without having to rewrite everything in Julia.

2.4.2 Scientific and Physics-based Machine Learning

Scientific machine learning (SciML) refers to the application of Machine Learning (ML) techniques to scientific problems, where the goal is not only to make predictions but also to gain insights into the underlying physical processes (Raissi et al., 2019; Rackauckas et al., 2020; Carleo et al., 2019). SciML involves the integration of domain-specific knowledge and physical models with data-driven techniques, and it has the potential to revolutionize many areas of science and engineering. In this thesis we explore the use of Physics-based machine learning (PBML) (Raissi and Karniadakis, 2021; Wu and Zhang, 2021) for a variety of applications.

Recent examples include a paper by (Raissi et al., 2019) that introduces a physics-informed neural network (PINN) framework for solving nonlinear partial differential equations, a paper by (Rackauckas et al., 2020) that proposes a universal differential equation (UDE) approach to scientific machine learning, and a review article by (Carleo et al., 2019) that discusses the use of machine learning in various fields of physics, including condensed matter physics, high-energy physics, and quantum physics. PBML has several advantages over purely data-driven approaches, including:

- Improved generalization: PBML models incorporate prior knowledge of the underlying physics, resulting in models that are more interpretable and generalizable. This enables the models to make accurate predictions even with limited training data.
- Incorporation of physical constraints: PBML models can be designed to incorporate physical constraints, such as conservation laws, which can help to ensure physically consistent predictions.

- Improved interpretability: PBML models are more interpretable than purely data-driven models since they are designed to incorporate physical principles. This can enable scientists and engineers to gain deeper insights into the underlying mechanisms of the systems they are studying.
- Reduced data requirements: PBML models require less training data than purely data-driven models since they leverage the physics-based priors, reducing the need for large datasets to train accurate models.
- Better extrapolation: PBML models are better equipped to extrapolate beyond the training data since they incorporate knowledge of the underlying physics, enabling them to make more accurate predictions in new and unseen scenarios.

Overall, PBML has several advantages over purely data-driven approaches, including improved generalization, reduced data requirements, better extrapolation, incorporation of physical constraints, and improved interpretability, making it a valuable tool for scientific and engineering applications.

NOTE: We should combine chapter on Physical Context with this one

We should end with some comments on work complete and work proposed as well as a tentative schedule

2.5 Water Quality

2.5.1 Properties of Aqueous Solutions

2.5.2 Reflectance Spectroscopy

NEEDS UPDATING/REWORDING:

The quantitative description of light absorption by a sample is based on the Beer-Lambert (or, more correctly, Beer-Lambert-Bouguer) law. In deriving this law, we consider that incident monochromatic light of intensity I_0 crosses an infinitesimal thickness, dl , of an absorbing

species of concentration c . The decrease in light intensity, dI , is proportional to the thickness of the sample, the concentration of the absorbing species and the incident light intensity

$$dI = -a(\lambda)CI d\ell \quad (2.1)$$

so that upon integrating, we find

$$I(\lambda) = I_0(\lambda) \exp(-a(\lambda)C\ell) \quad (2.2)$$

2.5.3 Solar Geometry

An explanation of relevant solar angles as well as their determination (i.e. the code I ported to Julia from Matlab script Dr. Lary supplied). We should also comment on the importance of

Use David's figure from his book to illustrate the solar geometry.

2.6 Air Quality

Exposure to air pollution has been linked to a wide range of health effects (Brook et al., 2010; Kelly and Fussell, 2011; Xu et al., 2017), including respiratory and cardiovascular diseases, cancer, and adverse birth outcomes. Further, physical sensing provides valuable data and the basis for international assessments such as the Intergovernmental Panel on Climate Change (IPCC), which seeks to assess the science related to climate change and its impacts on natural and human systems (Houghton et al., 1990, 1996, 2001; Solomon et al., 2007; Parry et al., 2007; Metz et al., 2007; Stocker et al., 2013; Field et al., 2014; Edenhofer et al., 2014; Masson-Delmotte et al., 2018; Friedlingstein et al., 2020; Huang et al., 2017).

2.6.1 Pollution and Particulate Matter

discuss sources, evolution, global trends, etc...

2.6.2 Indoor Air Quality

This statement serves as the foundation for The Clean Air in Buildings Challenge and this RFI. We completely agree with this premise, but we believe the ‘order of operations’ - among ventilation, filtration, and air cleaning - should be the primary focus of inquiry, and effective and safe air cleaning innovations should be a focus deserving full consideration. We can reconcile multiple, often competing goals by informing and empowering schools and commercial buildings to adopt a Clean First mentality, which we will explore in this response, as well as examine the ‘art of the possible’ by taking a critical look at the 21st century innovations we believe will be essential to the practical pursuit and achievement of The Clean Air in Buildings Challenge goals/objectives. We will demonstrate the necessary considerations of a Clean First approach as well as highlight its additional benefits in this response, which include but are not limited to:

- Reduced mechanical system energy consumption.
- Reductions in healthcare-acquired infections (HAIs) that are statistically significant in healthcare settings.
- Real-world testing/proof of lower microbiological burden (total and culturable bacteria, fungi, and their spores) in addition to improved ventilation/filtration alone.
- Reduced school absenteeism.
- Increased resiliency to future pandemics and communicable diseases.
- Reduction in airborne communicable disease and allergen loads.

The real challenges posed by global environmental change, combined with rising utility costs as a result of inflation and geopolitical shocks to energy supply chains, have created an implicit dichotomy between climate and public health goals. As Prof. Bahnfleth has said,

2.7 Physics of Chemical Reactions: Chemical Reaction Kinetics

2.7.1 Overview

Since the early successes of Newton’s descriptions of mechanics by means of simple forces acting on masses, scientists have sought to understand the dynamics of chemical reactions in terms of the detailed microphysics of molecular collisions. As we shall see, this approach can be utilized productively to justify the complicated temperature and pressure dependence of the reaction rate coefficients of many elementary reaction. However, even when considering the asymmetric structure of many molecules, and therefore, the dependence on orientation at the collision site, kinetic theory alone is unable to model reaction rates in all relevant temperature and pressure regimes. To do this, one can utilize the modern treatment of *Potential Energy Surface* (PES) theory together with the notion of short-lived, unstable intermediate reaction states to calculate reaction rate coefficient functions for specific reactants. *Ab initio* solution of the Schrodinger equation for the relevant nuclear geometries ($3N$ reaction coordinates for N atoms) together with scattering and spectroscopic methods as suggested by (Neumark, 1992) have led to significant improvements in our understanding of reaction dynamics.

However the computational complexity of this task makes it prohibitively expensive to perform at the scale required for our desired chemical mechanism which consists of many hundreds to thousands of reactants together with as many as 16,000 unique reactions. Therefore in the following discussion, we shall primarily utilize the kinetic theory to justify the functional form for *most* rate coefficients with some reference to the extensions made by PES theory. We note that in practice, kinetic evaluations such as the periodic reports from the NASA Jet Propulsion Laboratory (Burkholder et al., 2020) utilize (justified) empirical fits to provide suggested functional forms for reaction rate coefficients.

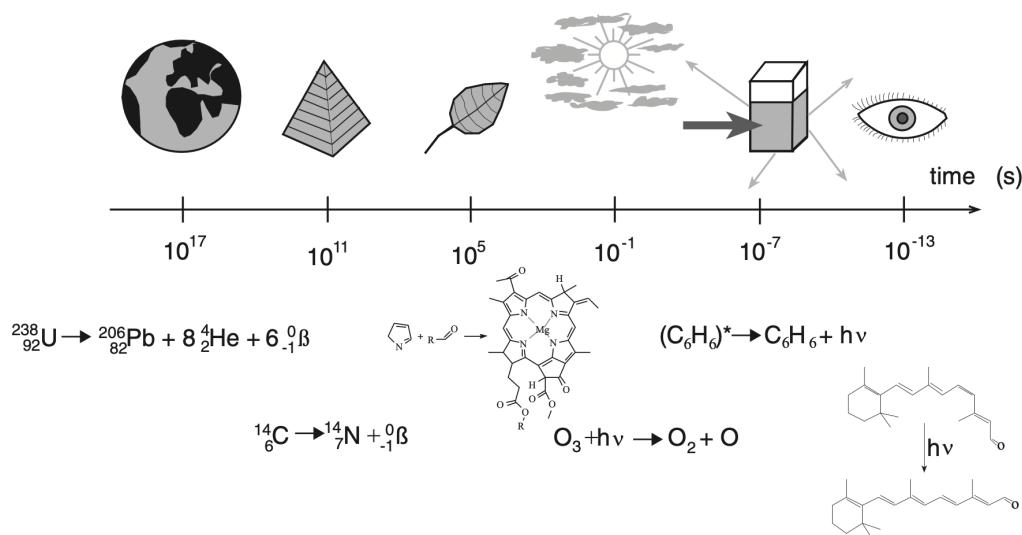


Figure 2.1. An illustration of the broad range of reaction time scales from the long-lived nuclear decay to rapid degradation of molecules by photolysis. Figure taken from (Arnaut and Burrows, 2006)

2.7.2 Chemical Equilibrium and the Law of Mass Action

Before outlining the dynamics involved during complex chains of chemical reactions, it is worth spending some time to consider how we should treat chemical equilibrium. Usually in these scenarios we can not hold the internal energy fixed due to interactions with the environment, but rather, the temperature and pressure can often be treated as so. For example, we may be interested in gas-phase reactions occurring in ambient indoor air at or near standard temperature and pressure. In such scenarios, one finds the relevant potential energy to be the Gibbs, given by

$$G = U - TS + PV, \quad (2.3)$$

which is minimized in equilibrium under constant temperature and pressure.

This equation leads to the convenient thermodynamic identity

$$dG = -SdT + VdP + \sum_i \mu_i dN_i \quad (2.4)$$

from which we may identify the *chemical potential* of the i^{th} species as

$$\mu_i = \left(\frac{\partial G}{\partial N_i} \right)_{T, P, N_{j \neq i}}. \quad (2.5)$$

The fact that each μ_i depends only on intensive state variables allows us to further simplify the relationship by considering what would happen were we to gradually increase the size of the system while maintaining the values of intensive parameters T , P , μ_i . The result is G must increase in direct proportion to the increase in each N_i , that is:

$$G = \sum_i \mu_i N_i. \quad (2.6)$$

From equation 2.6 it is clear that the μ_i can be understood as molecular *potentials* (i.e. chemical energy per molecule) in analogy to the notion of electric potential as a energy per unit charge.

For an ideal gas consisting of a single component we can combine equation 2.6 together with the identity

$$V = \left(\frac{\partial G}{\partial P} \right)_{T, N} \quad (2.7)$$

to obtain

$$\left(\frac{\partial \mu}{\partial P} \right)_{T, N} = \left(\frac{\partial}{\partial P} \frac{G}{N} \right)_{T, N} = \frac{1}{N} \left(\frac{\partial G}{\partial P} \right)_{T, N} = \frac{V}{N} = \frac{kT}{P} \quad (2.8)$$

so that by integration from a reference pressure, say $P_0 = 1$ atm, we obtain the handy expression for the chemical potential

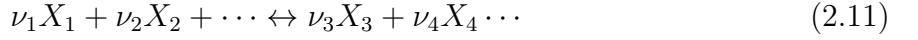
$$\mu(T, P) = \mu(T, P_0) + kT \ln(P/P_0) \quad (2.9)$$

which we shall use again momentarily.

Returning now to the notion of chemical equilibrium, recall that we must have $dG = 0$ so that G is minimized. At constant temperature and pressure, this means

$$0 = dG = -SdT + VdP + \sum_i \mu_i dN_i = \sum_i \mu_i dN_i \quad (2.10)$$

and therefore, a generic reaction of the form



with chemical species X_i and stoichiometric coefficients ν_i satisfies the condition that

$$\nu_1 \mu_1 + \nu_2 \mu_2 + \cdots = \nu_3 \mu_3 + \nu_4 \mu_4 + \cdots . \quad (2.12)$$

If we now make use of equation 2.9 with the identification of $\mu^0 = \mu(T, P_0)$, then we obtain

$$\sum_i^{\text{products}} \nu_i \mu_i^0 + \nu_i kT \ln(P_i/P_0) = \sum_j^{\text{reactants}} \nu_j \mu_j^0 + \nu_j kT \ln(P_j/P_0). \quad (2.13)$$

collecting terms involving the μ_k^0 to one side and multiplying through by Avogadro's constant, we obtain

$$RT \ln \left(\frac{\prod_j^{\text{reactants}} \left(\frac{P_j}{P_0} \right)^{\nu_j}}{\prod_i^{\text{products}} \left(\frac{P_i}{P_0} \right)^{\nu_i}} \right) = R \left(\sum_j^{\text{reactants}} \nu_j \mu_j^0 - \sum_i^{\text{products}} \nu_i \mu_i^0 \right) = \Delta G^0 \quad (2.14)$$

so that by exponentiation, we arrive at the simple expression:

$$\frac{\prod_j^{\text{products}} \left(\frac{P_j}{P_0} \right)^{\nu_j}}{\prod_i^{\text{reactants}} \left(\frac{P_i}{P_0} \right)^{\nu_i}} = \exp(-\Delta G^0/RT) \quad (2.15)$$

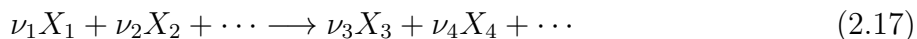
which through further application of the ideal gas law yields

$$\boxed{\frac{\prod_j^{\text{products}} (X_j)^{\nu_j}}{\prod_i^{\text{reactants}} (X_i)^{\nu_i}} = K_{\text{eq}}} \quad (2.16)$$

Here K_{eq} is a temperature dependent constant called the *equilibrium constant* for the reaction, and equation 2.16 is called the *law of mass action*. This expression indicates what we can expect to find if we allow our reactive system to proceed far enough to reach equilibrium. We shall later utilize this expression to perform a thermodynamic *sanity check* as is described in (Boldi, 1994).

2.7.3 Reaction Rate Laws

Having established the expected behavior at equilibrium, our task now is to establish the correct dynamical laws describing the variety of reactions which take place. To begin, let us consider again a generic chemical reaction of the form



To describe the dynamical process of a reaction, we can introduce a parameter ξ called the *reaction extent* such that at any time we have

$$\xi(t) = \frac{|N_i(t) - N_i(0)|}{\nu_i} \quad (2.18)$$

where $N_i(t)$ is the number of the i^{th} species and ν_i is the stoichiometric coefficient. The reaction rate is then easily understood as the rate of change of the reaction extent,

$$r := \frac{d\xi}{dt} = \frac{1}{\nu_i} \left| \frac{dN_i(t)}{dt} \right|. \quad (2.19)$$

Manipulating this expression to introduce the volume then leads us to

$$r(t) = \frac{1}{\nu_i} \left| \frac{dN_i}{dt} \frac{V}{V} \right| = \frac{V}{\nu_i} \left| \frac{d[X_i]}{dt} \right| \quad (2.20)$$

where $[X_i]$ denotes the concentration (number density) of the i^{th} species participating in the reaction. All this is to say that upon rearranging the expression, we find

$$\left| \frac{d[X_i]}{dt} \right| = \nu_i \frac{r(t)}{V} = \nu_i v, \quad (2.21)$$

or in words, the absolute change in concentration of the i^{th} species as a function of time is proportional the quantity $v = r(t)/V$ (called the reaction *velocity*) scaled by the stoichiometric coefficient ν_i of the i^{th} species. For the purposes of our modeling tasks, we must now establish appropriate forms for the reaction velocity in terms of the relevant thermodynamic variables (i.e. temperature, and pressure) and constituent concentrations.

To begin, let us consider a bimolecular reaction



The simplest approach to modeling the reaction velocity for reactions of this type is to make the assumption that *each molecular collision leads to a reaction*. From this perspective, should then be able to derive the reaction velocity using the established statistics of molecular velocities together with appropriate data for the size of each reactant.

Let us treat reacting species as *hard spheres* of radii r_A and r_B respectively, or in other words, the molecules are spheres which only interact by contact (no long range interactions). Then a collision will occurs if the separation d_{AB} satisfies

$$d_{AB} = r_A + r_B \quad (2.23)$$

NOTE: Add image of hard spheres here

To start, let's consider collisions where B are stationary and A is seen to move with velocity \mathbf{v}_A .

NOTE: Add image of hard spheres forming cylinder here

Then during a time dt , the molecule A sweeps out a cylindrical volume

$$dV = \pi d_{AB}^2 v_A dt \quad (2.24)$$

in which collisions may occur. Supposing we have a density N_B/V of species B , then the collision rate (collisions per second) for a single particle of A will be

$$\frac{dV}{dt} \frac{N_B}{V} = \frac{\pi d_{AB}^2 v_A N_B}{V} \quad (2.25)$$

and therefore, the reaction velocity for N_A particles with average velocity \bar{v}_A is

$$v = \frac{\pi d_{AB}^2 \bar{v}_A N_A N_B}{V^2} \quad (2.26)$$

if instead particles of both A and B move relative to each other, then their *relative* velocity is given by the law of cosines:

$$v_{AB}^2 = v_A^2 + v_B^2 - 2v_A v_B \cos(\theta) \quad (2.27)$$

which, since all directions are equally probable, yields an average relative velocity of

$$\overline{v_{AB}}^2 = \sqrt{\overline{v_A}^2 + \overline{v_B}^2}. \quad (2.28)$$

The relative velocity describes the motion of the reduced mass $\mu = m_A m_B / (m_A + m_B)$ and therefore under the standard Maxwellian velocity distribution, leads to

$$\overline{v_{AB}} = \sqrt{\frac{8kT}{\pi\mu}} \quad (2.29)$$

Combining everything together yields the reaction velocity

$$v = \pi d_{AB}^2 \frac{N_A}{V} \frac{N_B}{V} \sqrt{\frac{8kT}{\pi\mu}} = \pi d_{AB}^2 [A][B] \sqrt{\frac{8kT}{\pi\mu}} \quad (2.30)$$

which we might further simplify as

$$v = k[A][B] \quad (2.31)$$

$$k = \pi d_{AB}^2 \sqrt{\frac{8kT}{\pi\mu}} = \sigma \sqrt{\frac{8k_B T}{\pi\mu}} \quad (2.32)$$

where k is called the *reaction rate coefficient* and σ is the *reaction cross section*.

Excellent! We have discovered a couple of key features for the reaction velocity, namely, that it depends on a polynomial combination of reactant concentrations, *and* that there is clear temperature dependence due to the relationship between molecular velocities and temperature.

There are, however, some obvious limitations.

1. Not all collisions occur with orientations favorable for reaction (i.e. the hard sphere model isn't realistic for species).

2. Not all collisions will have enough energy for the reaction to proceed.

These limitations were well known, and in particular, Arrhenius suggested a competing function based on empirical studies of with

$$k = A \exp(-\alpha/T) \quad (2.33)$$

where α is some constant that depends on the reaction taking place. We can address the first point in a *hand-wavy* manner by simply including a geometric correction factor, $g \leq 1$, to account for the distribution of favorable orientations. To address the second point, it is worth establishing some minimal energy required for the reaction, E_a , by examining our Maxwellian distribution in closer detail. As we shall see, this will allow us to recover the exponential dependence suggested by Arrhenius.

The velocities \mathbf{v}_A and \mathbf{v}_B of each colliding pair of reactants define a plane. Therefore, for ease of calculation, we approximate the distribution of velocities near the collision site by the 2-dimensional Maxwellian speed distribution:

$$f(v) = \frac{\mu}{k_B T} v \exp\left(-\frac{\mu v^2}{k_B T}\right) \quad (2.34)$$

so that the fraction of particles with speed in the range $[v, v + dv]$ is

$$\frac{dN(v)}{N_{tot}} = f(v)dv = \frac{\mu}{k_B T} v \exp\left(-\frac{\mu v^2}{k_B T}\right) dv. \quad (2.35)$$

For an ideal gas under no external forces, we may identify the energy $\epsilon = \mu v^2/2$ so that $d\epsilon = \mu v dv$. Therefore, in energy space, this ratio becomes

$$\frac{dN(\epsilon)}{N_{tot}} = \frac{1}{k_B T} \exp(-\epsilon/k_B T) d\epsilon \quad (2.36)$$

If E_a is the minimum (*activation*) Energy required to engage the reactants, then the ratio of reactants with sufficient energy for the reaction to proceed is determined by integration to be

$$\left. \frac{N(\epsilon)}{N_{tot}} \right|_{\epsilon > E_a} = \int_{E_a}^{\infty} \frac{1}{k_B T} \exp(-\epsilon/k_B T) d\epsilon = \exp(-E_a/k_B T) \quad (2.37)$$

so that we may justify an additional correction to our reaction rate coefficient

$$k = g\pi d_{AB}^2 \sqrt{\frac{8k_B T}{\pi\mu}} \exp(-E_a/k_B T) \quad (2.38)$$

The final augmentation we can perform without before leaving collision theory behind is to observe that the cross-section σ was treated independently from the requirement of a minimum activation energy. With that in mind, suppose instead that a reaction cross section *only makes sense* if the reactants have the required minimum energy, that is

$$\sigma(\epsilon) = \begin{cases} \pi d_{AB}^2 & \epsilon > E_A \\ 0 & \text{otherwise} \end{cases} \quad (2.39)$$

If this is true, we can not keep the cross section outside of the ensemble average.

Allowing ourselves to utilize the full, three-dimensional Maxwellian speed distribution, we have

$$\begin{aligned} k &= \int_0^\infty \sigma(v) v f(v) dv \\ &= 4 \left(\frac{\mu}{2\pi k_B T} \right)^{3/2} \int_0^\infty \sigma(v) v \cdot v^2 \exp\left(-\frac{\mu v^2}{2k_B T}\right) dv \end{aligned} \quad (2.40)$$

which in terms of energy yields

$$\begin{aligned} k(T) &= \left(\frac{1}{\pi\mu} \right)^{1/2} \left(\frac{2}{k_B T} \right)^{3/2} \int_0^\infty \epsilon \sigma(\epsilon) \exp(-\epsilon/k_B T) d\epsilon \\ &= \left(\frac{1}{\pi\mu} \right)^{1/2} \left(\frac{2}{k_B T} \right)^{3/2} \int_{E_a}^\infty \epsilon \pi d_{AB}^2 \exp(-\epsilon/k_B T) d\epsilon \\ &= \pi d_{AB}^2 \sqrt{\frac{8k_B T}{\pi\mu}} \left(1 + \frac{E_a}{k_B T} \right) \exp(-E_a/k_B T) \end{aligned} \quad (2.41)$$

In summary, we have established that for bimolecular reactions, the (simple) theory of molecular collisions allows us to model the reaction velocity by

$$v = k(T)[A][B] \quad (2.42)$$

$$k(T) \approx \pi d_{AB}^2 \sqrt{\frac{8k_B T}{\pi \mu}} \left(1 + \frac{E_a}{k_B T}\right) \exp(-E_a/k_B T) \quad (2.43)$$

To derive a more accurate form for the rate coefficient k , we can resort to PES, simulation, and statistical sampling techniques (Partridge et al., 1993; Wu et al., 2006, for example). For our purposes, it is enough to have justified the kinds of functional dependence on temperature we can expect to encounter when collating available data on rate coefficients in the present literature.

2.8 Photolysis

Give an overview for photolysis (perhaps borrow from Dr. Lary's Textbook)

For experimental reasons, the wavelengths (λ) most commonly used for initiating photochemical processes vary between the ultraviolet (200 – 250 nm) and the near infrared (750 – 800 nm). Light at these wavelengths has an energy which corresponds roughly to 600 – 50 kJ mol⁻¹, which is very close to the energies of many chemical bonds.

2.9 Summary of Chemical Mechanism Kinetics

Give a summary of kinetics for each of the 3 reaction types. Then finish by writing the form of the combined differential equations for the entire system (and it's Jacobian)

CHAPTER 3

PHYSICAL SENSING

3.1 Hyperspectral Imaging

The following are notes from the Manolakis textbook that I originally kept here. NOTE: we will need to either make new figures or correctly cite these for attribution.

3.1.1 Hyperspectral Imaging Sensors

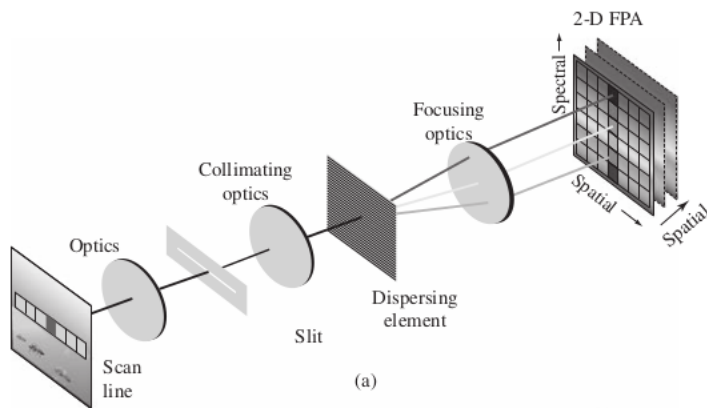
- *Hyperspectral Sensors* aka imaging spectrometers
 - scanning mechanism
 - imaging system
 - spectrometer
- 3 types of resolution
 1. spatial
 2. spectral
 3. radiant
 4. (temporal?)

3.1.2 Spectral-Spatial Data Collection and Organization

Data collected into *datacube* with 2 spatial dimensions, 1 spectral dimension

Different types of rigs:

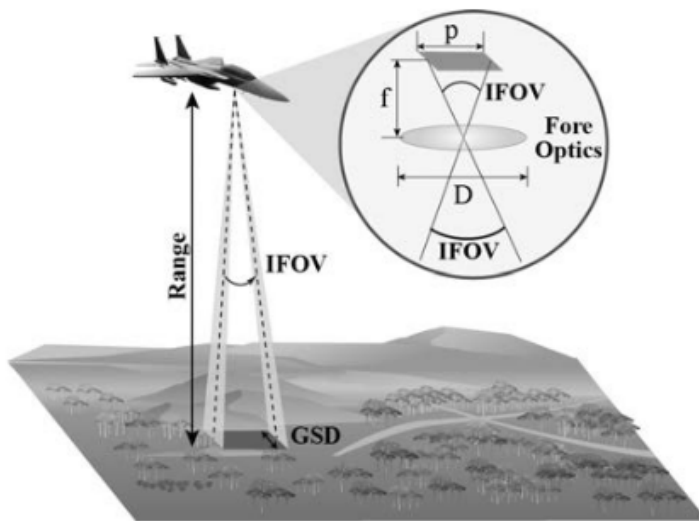
- Pushbroom scanner (ours)
- Staring System
- Fourier Transform Imaging Spectrometer (FTIS)



3.1.3 Spatial Sampling

- ground resolution elements are mapped to picture elements (pixels)
- IFOV: Instantaneous Field of View

- Cross track dimension: the projection of the long axis of the slit (i.e. the axis of the pushbroom sensors)
- Along track dimension: the direction accumulated by traveling
- Ground Sample Distance: physical size of projected pixel element



3.1.4 Spectral Sampling

- Recovery of spectral info is imperfect due to finite sampling
- Spectral Response Function: is the weighing function that describes the wavelengths that are transmitted to a particular spectral sample

3.1.5 Radiometric Sampling

- detector transforms radiant power to electrical signal
- electrical signal converted to number via analog-to-digital converter
- photon detectors

3.1.6 Signal Considerations

Strength of signal is determined by:

- Terrain composition affects amount of radiant energy reflected/emitted from ground resolution element
- Range: Intensity drops off by inverse square law. Further you are away, the worse the signal
- Spectral Bandwidth: output signal of detector element is proportional to spectral bandwidth of the detector
- Instantaneous Field of View: Decreasing IFOV increases spatial resolution but weakens the signal
- Dwell Time: the time required to sweep the IFOV across the ground resolution element, i.e. the time-on-pixel. Longer dwell time \rightarrow more accumulated photons \rightarrow more signal.

3.2 Remote Sensing

mention different types of satellite data platforms (mostly optical), differences in orbits, coverage, etc... Also good to discuss the increasing use of drones for a variety of applications including intelligent agriculture, geophysics, mapping, etc...

3.2.1 Infrared Sensing Phenomenology

Main passive sources of EM radiation for remote sensing are light emitted by the sun and the self-emission via black-body radiation of objects due to their temperature.

3.2.2 Sources of Infrared Radiation

- **spectral radiant exitance** power per unit area emitted by the sun. We can treat this as a black body with temperature $5800K$, maximum emittance at $\lambda = 0.50 \mu m$.
- The Earth is $300K$ with maximum spectral radiant emittance at $\lambda = 9.7 \mu m$. This is known as the **thermal infrared**

3.2.3 Atmospheric Propagation

- Key parameter is the **path length** of atmosphere traveled through before it arrives at the remote sensing system. Main effects are:
 - **Atmospheric Scattering**: diffusion of radiation by particles in the atmosphere
 - **Absorption**
- Useful remote sensing spectral regions are obtained via the **Transmission Spectrum**.
 - **Reflective Range**: $0.35 - 2.5 \mu m$. Dominated by solar illumination
 - **Water Absorption**: $0.2 - 2.5 \mu m$.
- **Atmospheric Windows**: Regions of low atmospheric absorption

3.2.4 Reflectance and Emissivity Spectra

There are three processes that occur when EM radiation meets an interface:

1. **Reflection**: Solar illumination dominates here. Consequently, this part of the spectrum is used to characterize the surface
 - **Specular Reflectors**: Flat surfaces that act like mirrors, i.e. $\theta_i = \theta_r$.
 - **Diffuse (Lambertian) Reflectors**: Rough surfaces that reflect uniformly in all directions.

- **Real Reflectors:** Somewhere between the specular and diffuse.

2. Absorption

3. Transmission

- Fractions vary as a function of λ
- Remote sensing usually cares about *diffuse* reflectors because this is the dominant type of most materials (water being an exception).
- Reflectance of a material is characterized by its **Reflectance Spectrum**, that is, the percent of incident light reflected as a function of wavelength.
 - Dips in reflectance spectrum are called **absorption features**
 - Peaks are called **Reflectance Peaks**
- **Emissivity Spectrum:** The ratio of radiant emittance at a given temperature to the radiant emittance of a black body at the same temperature.

3.3 Coordinated Robotic Teams

- Remote Sensing: data acquisition, processing, and interpretation of images, and related data, obtained from aircraft and satellites that record the interaction between matter and electromagnetic radiation
- Source: the source of electromagnetic radiation, e.g. the sun, black-body radiation, microwave radar, etc...
- Atmospheric Radiation: The EM radiation propagating through the atmosphere. Moderated by various processes including absorption and scattering

- Earth's Surface Interactions: Amount and spectral distribution of radiation emitted/reflected by the earth's surface. This depends on
 - physical properties of the matter
 - wavelength of EM radiation that is sensed

3.4 In-situ Chemical Sensing in Water

Here we can give an overview of the sensors utilized on the boat for the robot team studies... We should include any information about the uncertainties. We can also comment on their use in fresh v.s. salt water environments, e.g. we should discuss fluorimeters, sonar, and any other relevant sensors from the boat

3.5 Low Cost Sensors for (Outdoor) Air Quality

The outdoor part here is optional. Here I seek to provide specific details of the classes of low cost sensors used for PM (optical particle counters based on Mie scattering), thermistors (for temperature), humidity sensors, pressure sensors, gas sensors, etc... This can be a rough outline for those that specifically are used in my research projects.

3.6 The HEART Chamber

CHAPTER 4

MACHINE LEARNING METHODS

NOTE: Good way to start is with the question: What is a model? We can discuss the differences between mechanistic models (i.e. physics equations), their free parameters (constants of nature) and other types of models such as the non-parametric, nonlinear models used in machine learning. Further, we can discuss how machine learning models often display have the desirable trait of being a **universal approximator**. This will naturally lead to a discussion of function expansions (Taylor, Fourier, other polynomial expansions, etc...) and how they scale poorly (exponential) with increasing dimension. Machine learning models essentially allow us to do the same thing: perform a function expansion given data but in a way that can scale well to arbitrary dimension (features) of data. This allows us to build predictive models without necessarily needing to prescribe *all* of the physics. From another perspective, this framework allows us to incorporate our physics knowledge from well-behaved or linearized domains in order to *fit the residual* behavior with a sophisticated data-driven approach.

discuss role in science in particular (i.e. calibration, modeling, etc...)

discuss pushback against use in science and need for methods which simultaneously provide uncertainty bounds. Also describe types of ML e.g. supervised, unsupervised, generative, etc...

this is a good place for the Chihuahua vs Muffin meme... and use this to motivate incorporating physical knowledge into the machine learning process as a key feature for scientific applications... we have more constraints!

also discuss statistical v.s. deep learning

4.1 Adjoint Methods for Optimization

The fundamental task of machine learning is to build data-driven models. To do this we must *fit* model parameters by taking advantage of the available training data. In many cases, this is straight forward: the sensitivities of model outputs to changes in model parameters can be backpropagated by application of the chain rule through each layer of our model. However, this naive propagation can become prohibitively expensive as we increase the size and complexity of our models. To efficiently determine these sensitivities and thereby enable the optimization of complicated models, we can instead turn the problem on it's side and consider a parallel *adjoint* system. Due to the utility of this technique for enabling the optimization of models involving implicit linear and nonlinear relationships and even differential equations, we shall begin our discussion of machine learning with a handful of these techniques which will be utilized in the remainder of this work.

4.1.1 Adjoint Methods for Linear Systems

To begin, suppose a component of our model is given by the θ -parameterized linear system

$$A(\theta)u = b(\theta). \quad (4.1)$$

If we wish to find the ideal θ subject to some loss function $J(u)$ we must obtain the gradients $dJ/d\theta$ which upon expansion yields

$$\frac{dJ}{d\theta} = \frac{\partial J}{\partial \theta} + \frac{\partial J}{\partial u} \frac{du}{d\theta}. \quad (4.2)$$

The Jacobian $du/d\theta$ will be challenging to compute and depends on solving the above implicit system:

$$\begin{aligned} \frac{d}{d\theta}(Au) &= \frac{d}{d\theta}b \\ \left(\frac{dA}{d\theta}\right)u + A\left(\frac{du}{d\theta}\right) &= \frac{db}{d\theta} \end{aligned}$$

which with some slight abuse of notation in the, yields

$$\frac{du}{d\theta} = A^{-1} \left(\frac{db}{d\theta} - \frac{dA}{d\theta} u \right) \quad (4.3)$$

and therefore

$$\frac{dJ}{d\theta} = \frac{\partial J}{\partial \theta} + \frac{\partial J}{\partial u} A^{-1} \left(\frac{db}{d\theta} - \frac{dA}{d\theta} u \right) \quad (4.4)$$

must require p linear solves if there are p parameters in θ . Can we do better?

If we perform a clever *re-bracketing* so that

$$\frac{dJ}{d\theta} = \frac{\partial J}{\partial \theta} + \left(\frac{\partial J}{\partial u} A^{-1} \right) \left(\frac{db}{d\theta} - \frac{dA}{d\theta} u \right) \quad (4.5)$$

then we can see that the vector $\frac{\partial J}{\partial u} A^{-1}$ is the solution to *some* problem, say

$$\lambda^T A = \frac{\partial J}{\partial u} \quad (4.6)$$

then, we need only solve

$$A^T \lambda = \left(\frac{\partial J}{\partial u} \right)^T \quad (4.7)$$

once! This leaves us with the following procedure:

1. Solve the system $Au = b$ for u
2. Solve the adjoint system $A^T \lambda = \left(\frac{\partial J}{\partial u} \right)^T$ for λ
3. compute $\frac{dJ}{d\theta} = \frac{\partial J}{\partial \theta} + \lambda^T \left(\frac{db}{d\theta} - \frac{dA}{d\theta} u \right)$

4.1.2 Adjoint Methods for Nonlinear Systems

Having demonstrated an effective strategy for computing the gradients of cost functions with respect to underlying linear systems, a natural next step is to ask: can we do the same for nonlinear systems? Suppose now that we instead we find a component of our model to be of the form

$$f(u; \theta) = 0 \quad (4.8)$$

Then the gradient of some loss function $J(u; \theta)$ which depends indirectly on the solution of this system will be

$$\frac{dJ}{d\theta} = \frac{\partial J}{\partial \theta} + \frac{\partial J}{\partial u} \frac{du}{d\theta} \quad (4.9)$$

while the nonlinear system satisfies

$$\frac{df}{d\theta} = \frac{\partial f}{\partial u} \frac{du}{d\theta} + \frac{\partial f}{\partial \theta} = 0. \quad (4.10)$$

Therefore, the linear system that resulting from the differentiation procedure leads to

$$\frac{du}{d\theta} = - \left(\frac{\partial f}{\partial u} \right)^{-1} \frac{\partial f}{\partial \theta} \quad (4.11)$$

which we may substitute to find

$$\frac{dJ}{d\theta} = \frac{\partial J}{\partial \theta} - \left(\frac{\partial J}{\partial u} \left(\frac{\partial f}{\partial u} \right)^{-1} \right) \frac{\partial f}{\partial \theta} \quad (4.12)$$

so that again, by cleverly re-bracketing the above equations we may identify that

$$\frac{\partial J}{\partial u} \left(\frac{\partial f}{\partial u} \right)^{-1}$$

is the solution to *some* linear system of the form

$$\lambda^T \left(\frac{\partial f}{\partial u} \right) = \frac{\partial J}{\partial u}. \quad (4.13)$$

Therefore, we have arrived at a procedure similar to the linear case which is as follows:

1. Solve $f(u; \theta) = 0$ for u by your favorite algorithm.
2. Solve $\left(\frac{\partial f}{\partial u} \right)^T \lambda = \left(\frac{\partial J}{\partial u} \right)^T$ for the adjoints λ .
3. Compute the gradient $\frac{dJ}{d\theta} = \frac{\partial J}{\partial \theta} - \lambda^T \left(\frac{\partial f}{\partial \theta} \right)$.

4.1.3 Adjoint Methods for ODEs

To complete the discussion, suppose that we instead seek to fit a model described by an Ordinary Differential Equations (ODE) initial-value problem (IVP) of the form

$$\begin{cases} \frac{du}{dt} = f(u, t; \theta) \\ u_0 = u(t = 0) \end{cases} \quad (4.14)$$

where $u \in \mathbb{R}^n$ denotes the time-dependent state vector, and f is a function of the state u , time t , and some number of parameters θ .

To *fit* such a model to data, we need to be able to compute the sensitivity of some cost function $J(u)$ to the parameters θ , that is, we need the gradient $dJ/d\theta$. Obtaining these sensitivities by direct forward-mode automatic differentiation or with tape/tracer based backpropagation demands we be able to establish the derivatives for any class of ODE integrator we want to choose. This task is further complicated if we desire to use modern differential equation solver suites, like those provided by the `DifferentialEquations.jl` Julia library, which employ complicated adaptive stepping schemes. How could we possibly know *a priori* how many function evaluations will be needed to produce the output for each ODE step, and thereby how to compute the partial derivatives? One approach is to establish language-wide automatic differentiation capability for generic computer code (dubbed ∂P) (Innes et al., 2019). An alternative approach, which we shall describe here, is to develop a set of adjoint differential equations whose solution we can use to compute the desired derivatives.

Let us suppose our loss function can be written in the form

$$J(u; \theta) = \int_0^T g(u; \theta) dt \quad (4.15)$$

where g is some function like the quadratic loss $u^T R u$ of 4d-var. To find our desired sensitivities, we can utilize the method of Lagrange multipliers by introducing a Lagrangian of

the form

$$\begin{aligned}\mathcal{L}(u, \lambda; \theta) &:= J(u; \theta) + \int_0^T \lambda^T \left(f - \frac{du}{dt} \right) dt \\ &= \int_0^T \left[g(u; \theta) + \lambda^T(t) \left(f - \frac{du}{dt} \right) \right] dt\end{aligned}\tag{4.16}$$

The λ are the so-called Lagrange multipliers, and the integrand, $f - du/dt$, is a clever way of adding 0 so that $d\mathcal{L}/d\theta = dJ/d\theta$. Evaluating this derivative, we find

$$\frac{d\mathcal{L}}{d\theta} = \int_0^T \left\{ \left(\frac{\partial g}{\partial \theta} + \frac{\partial g}{\partial u} \frac{du}{d\theta} \right) + \lambda^T(t) \left[\frac{\partial f}{\partial \theta} + \frac{\partial f}{\partial u} \frac{du}{d\theta} - \frac{d}{dt} \frac{du}{d\theta} \right] \right\} dt. \tag{4.17}$$

The difficult term to compute is $du/d\theta$ and so we group the terms so that

$$\frac{d\mathcal{L}}{d\theta} = \int_0^T \left[\frac{\partial g}{\partial \theta} + \lambda^T(t) \frac{\partial f}{\partial \theta} + \left(\frac{\partial g}{\partial u} + \lambda^T(t) \frac{\partial f}{\partial u} - \lambda^T(t) \frac{d}{dt} \right) \frac{du}{d\theta} \right] dt. \tag{4.18}$$

We now observe that if we pick the $\lambda^T(t)$ such that

$$\lambda^T(t) \frac{\partial f}{\partial \theta} + \left(\frac{\partial g}{\partial u} + \lambda^T(t) \frac{\partial f}{\partial u} - \lambda^T(t) \frac{d}{dt} \right) \frac{du}{d\theta} = 0, \tag{4.19}$$

we will not have to compute this pesky term at all! To proceed, let's apply integration by parts to move the time derivative of the final term and simplify the matter:

$$\begin{aligned}\int_0^T -\lambda^T \frac{d}{dt} \frac{du}{d\theta} dt &= -\lambda^T \frac{du}{d\theta} \Big|_0^T + \int_0^T \frac{d\lambda^T}{dt} \frac{du}{d\theta} dt \\ &= \lambda^T(0) \frac{du}{d\theta}(0) - \lambda^T(T) \frac{du}{d\theta}(T) + \int_0^T \dot{\lambda}^T \frac{du}{d\theta} dt.\end{aligned}\tag{4.20}$$

Plugging this back in to our previous expression yields

$$\frac{d\mathcal{L}}{d\theta} = \int_0^T \left[\frac{\partial g}{\partial \theta} + \lambda^T \frac{\partial f}{\partial \theta} + \left(\frac{\partial g}{\partial u} + \lambda^T \frac{\partial f}{\partial u} + \dot{\lambda}^T \right) \frac{du}{d\theta} \right] dt + \lambda^T(0) \frac{du}{d\theta}(0) - \lambda^T(T) \frac{du}{d\theta}(T). \tag{4.21}$$

Great! In order to make the pesky term disappear, it suffices to find the $\lambda^T(t)$ such that

$$\begin{cases} \frac{\partial g}{\partial u} + \lambda^T \frac{\partial f}{\partial u} + \frac{d\lambda^T}{dt} = 0 \\ \lambda(T) = 0 \end{cases} \tag{4.22}$$

which amounts simply solving a second differential equation from the ending time T back to $t = 0$. Transposing the above, we can obtain the new ODE in the standard form:

$$\boxed{\begin{cases} \frac{d\lambda}{dt} = -\left(\frac{\partial f}{\partial u}\right)^T \lambda - \left(\frac{\partial g}{\partial u}\right)^T \\ \lambda(T) = 0 \end{cases}} \quad (4.23)$$

Better yet, this system is linear in λ even though the original differential equation may be fully nonlinear with no restrictions on f ! To summarize, the procedure for determining the desired gradients is as follows:

1. Solve the ODE system given by $du/dt = f(u, t; \theta)$ forward in time from $t = 0$ to $t = T$ to obtain the solution $u(t)$.
2. Solve the Adjoint ODE system given by $d\lambda/dt = -(\partial f/\partial u)^T \lambda - (\partial g/\partial u)^T$ from $t = T$ to $t = 0$ to obtain the adjoints $\lambda^T(t)$
3. Compute the gradient $dJ/d\theta = d\mathcal{L}/d\theta = \int_0^T [(\partial g/\partial \theta) + \lambda^T(t)(\partial f/\partial \theta)]dt + \lambda(0)du_0/d\theta$ by quadrature.

4.2 Exploratory Data Analysis

4.2.1 Correlation

4.2.2 Mutual Information

4.3 Model Training Methodology

4.3.1 Data Sampling

Dr. Lary's method (from Gaussian Process Code) for representative sampling to reduce data size

- Testing holdout set

- k-fold Cross Validation

We should use this subsection to discuss why k-fold cross validation is ideal where data size/time permits so that you can get a sense of each model's dependence on the input data. Similarly, it is important we end the discussion by pointing out that the final **production** model is to be trained on the *entire* dataset after we have evaluated which model is best to use and the ideal hyperparameters, etc...

4.3.2 Model Evaluation

Here we should discuss how we evaluate models, e.g. scatter plots, quantile-quantile plots, histograms, confusion matrices, various correctness metrics, etc...

4.3.3 Hyperparameter Selection

Discuss grid search vs random search vs Bayesian Optimization, etc...

4.3.4 Occam's Razor and Feature Importance Ranking

I should point out my contributions to various julia packages in MLJ for doing feature importance determination here

Linear Regression

Tree Based Methods

Neural Networks

Perhaps explore ideas from Chris Olah's blog??? From a perspective of basic function composition... as in Rackauckas's blog

Shapley Values

Here we can discuss Shapley Values as a model agnostic method for feature importance rankings with the caveat that their computation can take quite a while...

4.4 Supervised Regression Techniques

4.4.1 Neural Networks

4.4.2 Gaussian Process Regression

Based on my notes from this repo

The following is based on the book **Gaussian Processes for Machine Learning** by Carl Edward Rasmussen and Christopher K. I. Williams (Rasmussen et al., 2006). You can find the free online book [here](#).

To explain/derive the Gaussian Process model for regression, let's first consider a motivating example: **Linear Regression**. We will use this guiding example to derive GRP from a *weight space view*. After this derivation, we will suggest a simpler, but more abstract derivation using a *function space view*. First let's set up some data we can use for training:

The Weight Space View

We consider a dataset \mathcal{D} with n observations

$$\mathcal{D} = \left\{ (\mathbf{x}_i, y_i) \mid i = 1, \dots, n \right\} \quad (4.24)$$

- \mathbf{x}_i is the i^{th} D-dimensional input (feature) vector
- y_i is the i^{th} target

Linear regression is easily understood in terms of *linear algebra*. We therefore collect our dataset \mathcal{D} into a $D \times n$ dimensional **Design Matrix**. Note that we have used a transposed

definition (features are rows, records are columns) as Julia is a column-major language (like Matlab and Fortran).

$$X := \begin{pmatrix} \vdots & \vdots & & \vdots \\ \mathbf{x}_1 & \mathbf{x}_2 & \dots & \mathbf{x}_n \\ \vdots & \vdots & & \vdots \end{pmatrix} \quad (4.25)$$

and our targets into a target vector

$$\mathbf{y} := (y_1, \dots, y_n) \quad (4.26)$$

so that the full training set becomes

$$\mathcal{D} := (X, \mathbf{y}) \quad (4.27)$$

Standard linear regression is a model of the form

$$f(\mathbf{x}) = \mathbf{x}^T \mathbf{w} \quad (4.28)$$

where \mathbf{w} is the D -dimensional vector of weights. By minimizing the mean-squared-error between our model and targets, one can show that the optimal weights are given by

$$\mathbf{w} = (X X^T)^{-1} X \mathbf{y} \quad (4.29)$$

Note, this can also be easily obtained geometrically by finding the vector with the shortest distance to the hyperplane defined by the column space of X . This corresponds to solving the normal equations.

$$X X^T \mathbf{w} = X \mathbf{y} \quad (4.30)$$

The following demonstrates this procedure on a simple dataset

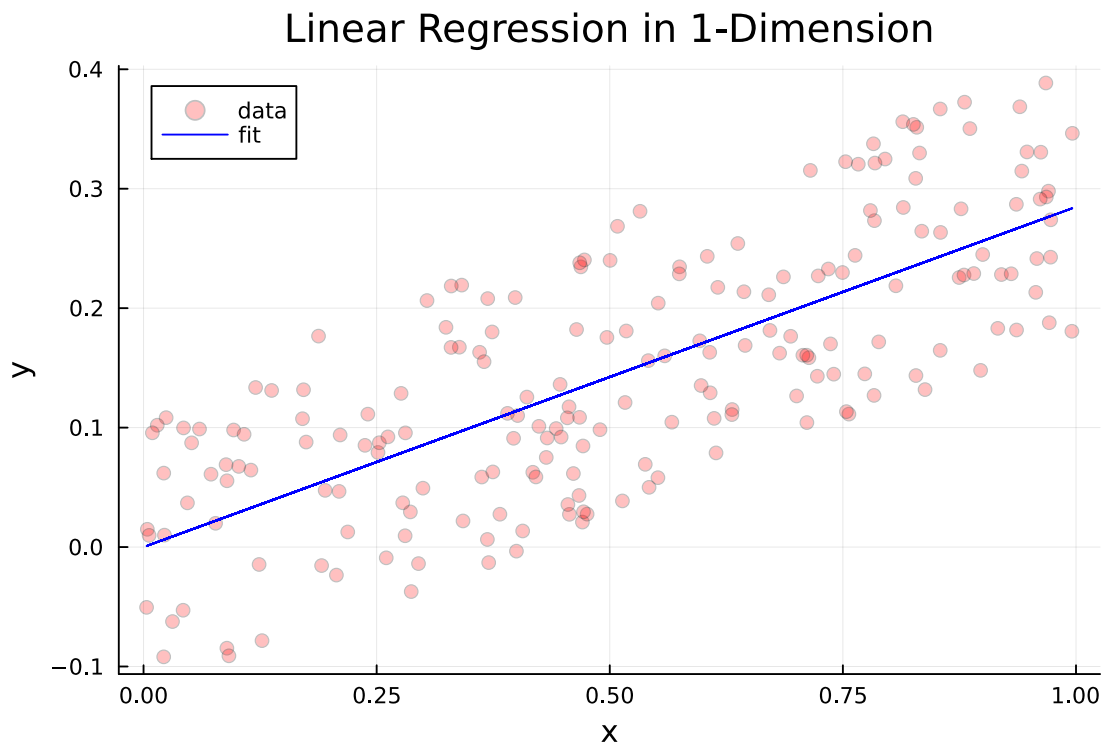


Figure 4.1. A standard linear regression fit on some noisy data.

We can also fit a y-intercept (aka *bias*) by augmenting the design matrix X to contain an extra row with all 1's, i.e.

$$X[D + 1, :] = (1, \dots, 1) \quad (4.31)$$

Standard linear regression assumes that are data \mathcal{D} are perfect but we can clearly see that the above data are noisy. To account for this, we need to make our model *Bayesian* by augmenting it to consider the measurement error. We define

$$f(\mathbf{x}) = \mathbf{x}^T \mathbf{w} \quad (4.32)$$

$$\mathbf{y} = f(\mathbf{x}) + \epsilon \quad (4.33)$$

$$\epsilon \sim \mathcal{N}(0, \sigma_n^2) \quad (4.34)$$

or, in words, our observed values differ from the *truth* by identically, independently, distributed Gaussian noise with mean 0 and variance σ_n^2 . The assumption that the noise is

i.i.d. is critical because it allows us to simplify the *likelihood* function by separating out each individual contribution by our datapoints:

$$p(\mathbf{y}|X, \mathbf{w}) := \prod_i^n p(\mathbf{y}_i|\mathbf{x}_i, \mathbf{w}) \quad (4.35)$$

$$= \prod_i^n \frac{1}{\sqrt{2\pi\sigma_n^2}} \exp\left(-\frac{(\mathbf{y}_i - \mathbf{x}_i^T \mathbf{w})^2}{2\sigma_n^2}\right) \quad (4.36)$$

$$= \frac{1}{(2\pi\sigma_n^2)^{n/2}} \exp\left(-\frac{1}{2\sigma_n^2} |\mathbf{y} - X^T \mathbf{w}|^2\right) \quad (4.37)$$

$$= \mathcal{N}(X^T \mathbf{w}, \sigma_n^2 I) \quad (4.38)$$

To perform inference with this updated model, we apply Baye's Rule, that is:

$$p(\mathbf{w}|\mathbf{y}, X) = \frac{p(\mathbf{y}|X, \mathbf{w})p(\mathbf{w})}{p(\mathbf{y}|X)} \quad (4.39)$$

where

- $p(\mathbf{w}|\mathbf{y}, X)$ is the *posterior distribution*
- $p(\mathbf{y}|X, \mathbf{w})$ is the *likelihood*
- $p(\mathbf{w})$ is the *prior distribution*
- $p(\mathbf{y}|X)$ is the *marginal likelihood*, i.e. the normalization constant

It is now that the utility of choosing gaussian distributions for our likelihood and prior becomes clear. We have

$$p(\mathbf{w}|\mathbf{y}, X) \propto \exp\left(-\frac{1}{2\sigma_n^2} (\mathbf{y} - X^T \mathbf{w})^T (\mathbf{y} - X^T \mathbf{w})\right) \exp\left(-\frac{1}{2} \mathbf{w}^T \Sigma_p^{-1} \mathbf{w}\right) \quad (4.40)$$

Taking the log and expanding leads to

$$\log(p(\mathbf{w}|\mathbf{y}, X)) = \frac{1}{2} \left[\frac{1}{\sigma_n^2} \mathbf{y}^T \mathbf{y} - \frac{1}{\sigma_n^2} \mathbf{y}^T X^T \mathbf{w} - \frac{1}{\sigma_n^2} \mathbf{w}^T X \mathbf{y} + \frac{1}{\sigma_n^2} \mathbf{w}^T X X^T \mathbf{w} + \mathbf{w}^T \Sigma_p^{-1} \mathbf{w} \right] \quad (4.41)$$

$$= \frac{1}{2} \left[\mathbf{w}^T \left(\frac{1}{\sigma_n^2} X X^T + \Sigma_p^{-1} \right) \mathbf{w} - \left(\frac{1}{\sigma_n^2} \mathbf{y}^T X^T \right) \mathbf{w} - \mathbf{w}^T \left(\frac{1}{\sigma_n^2} X \mathbf{y} \right) + \mathbf{y}^T \frac{1}{\sigma_n^2} \mathbf{y} \right] \quad (4.42)$$

$$= \mathbf{w}^T A \mathbf{w} - B^T \mathbf{w} - \mathbf{w}^T B + C \quad (4.43)$$

where we have defined

$$A := \frac{1}{\sigma_n^2} X X^T + \Sigma_p^{-1} \quad (4.44)$$

$$B := \frac{1}{\sigma_n^2} X \mathbf{y} \quad (4.45)$$

$$C := \mathbf{y}^T \frac{1}{\sigma_n^2} \mathbf{y} \quad (4.46)$$

Now we can complete the square so that

$$\mathbf{w}^T A \mathbf{w} - B^T \mathbf{w} - \mathbf{w}^T B + C = (\mathbf{w} - \bar{\mathbf{w}})^T A (\mathbf{w} - \bar{\mathbf{w}}) + K \quad (4.47)$$

leading to

$$\bar{\mathbf{w}} = A^{-1} B = \frac{1}{\sigma_n^2} \left(\frac{1}{\sigma_n^2} X X^T + \Sigma_p^{-1} \right)^{-1} X \mathbf{y} \quad (4.48)$$

$$K = C - \bar{\mathbf{w}}^T A \bar{\mathbf{w}} \quad (4.49)$$

Since K does not depend on \mathbf{w} directly, it may be absorbed into the normalization of $p(\mathbf{w}|\mathbf{y}, X)$. Thus we are left with

$$p(\mathbf{w}|\mathbf{y}, X) = \mathcal{N} \left(\bar{\mathbf{w}} = \frac{1}{\sigma_n^2} A^{-1} X \mathbf{y}, \Sigma = A^{-1} \right) \quad (4.50)$$

$$A = \frac{1}{\sigma_n^2} X X^T + \Sigma_p^{-1} \quad (4.51)$$

This result gives us the gaussian distribution over the space of possible parameter vectors \mathbf{w} . To use this distribution to make predictions, consider a newly supplied testpoint \mathbf{x}_* . We want to find

$$p(y_*|\mathbf{x}_*, \mathbf{y}, X) \quad (4.52)$$

We do this by marginalizing over our weight distribution, i.e.

$$p(y_*|\mathbf{x}_*, \mathbf{y}, X) = \int_{\mathbf{w}} p(y_*|\mathbf{x}_*, \mathbf{w})p(\mathbf{w}|\mathbf{y}, X)d\mathbf{w} \quad (4.53)$$

If we make the further assumption that testing points are i.i.d. gaussian distributed, we see that this integral is the product of two gaussians and therefore is also a gaussian. To find the mean and covariance of the predictive distribution, we check

$$\bar{y}_* = \mathbb{E}[y_*] = \mathbb{E}[\mathbf{x}_*^T \mathbf{w}] = \mathbf{x}_*^T \mathbb{E}[\mathbf{w}] = \mathbf{x}_*^T \bar{\mathbf{w}} \quad (4.54)$$

$$\text{Cov}(y_*) = \mathbb{E}[(y_* - \bar{y}_*)(y_* - \bar{y}_*)^T] \quad (4.55)$$

$$= \mathbb{E}[(\mathbf{x}_*^T \mathbf{w} - \mathbf{x}_*^T \bar{\mathbf{w}})(\mathbf{x}_*^T \mathbf{w} - \mathbf{x}_*^T \bar{\mathbf{w}})^T] \quad (4.56)$$

$$= \mathbb{E}[\mathbf{x}_*^T (\mathbf{w} - \bar{\mathbf{w}})(\mathbf{w} - \bar{\mathbf{w}})^T \mathbf{x}_*] \quad (4.57)$$

$$= \mathbf{x}_*^T \mathbb{E}[(\mathbf{w} - \bar{\mathbf{w}})(\mathbf{w} - \bar{\mathbf{w}})^T] \mathbf{x}_* \quad (4.58)$$

$$= \mathbf{x}_*^T \text{Cov}(\mathbf{w}) \mathbf{x}_* \quad (4.59)$$

$$= \mathbf{x}_*^T A^{-1} \mathbf{x}_* \quad (4.60)$$

so that

$$\boxed{p(y_*|\mathbf{x}_*, \mathbf{y}, X) = \mathcal{N}(\mathbf{x}_*^T \mathbf{w}, \mathbf{x}_*^T A^{-1} \mathbf{x}_*)} \quad (4.61)$$

Let's now take a break from our Bayesian regression discussion and return to the standard linear regression model for a moment. The key drawback of linear models like this is, of course, that they're *linear*!. Considering that many (most) *interesting* relationships are non-linear, how can we extend our simple linear model to enable us to perform complicated non-linear fits?

In the parlance of machine learning, the simple solution is to do feature engineering. If our initial feature vector is

$$\mathbf{x} = (x_1, \dots, x_n) \quad (4.62)$$

we can use our *expertise* to concoct new combinations of these features to produce the augmented vector

$$\tilde{\mathbf{x}} = (x_1, \dots, x_n, x_1^2, \sin(x_2), x_5 x_7 / x_4, \dots) \quad (4.63)$$

As an example, a linear classifier is unable to distinguish points inside a circle from those outside just from the (x, y) coordinates alone. Augmenting the feature vector to include the squared radius $x^2 + y^2$ as a new feature removes this obstacle. This works because the *linear* part of linear regression only refers to the fact that our model takes *linear combinations* of feature variables to produce its output. There is no restriction that the features themselves need to be independent variables! This same idea is what makes methods like SINDy work which use libraries of polynomial combinations of base features.

Constructing new features is often more art than science. To standardize the process, let's abstract the mapping from the original feature vector \mathbf{x} to the augmented vector $\tilde{\mathbf{x}}$. This is accomplished via the projection map $\phi : \mathbb{R}^D \rightarrow \mathbb{R}^N$ where

$$\mathbf{x} \mapsto \tilde{\mathbf{x}} = \phi(\mathbf{x}) \quad (4.64)$$

The result is that our linear model updates to become

$$f(\mathbf{x}) := \phi(\mathbf{x})^T \mathbf{w} \quad (4.65)$$

where the weight vector has gone from D dimensional to N dimensional. Similarly, the normal equations for \mathbf{w} update to become

$$\mathbf{w} = (\Phi \Phi^T)^{-1} \Phi \mathbf{y} \quad (4.66)$$

where $\Phi = \phi(X)$ is the $N \times n$ matrix resulting from applying ϕ columnwise to X .

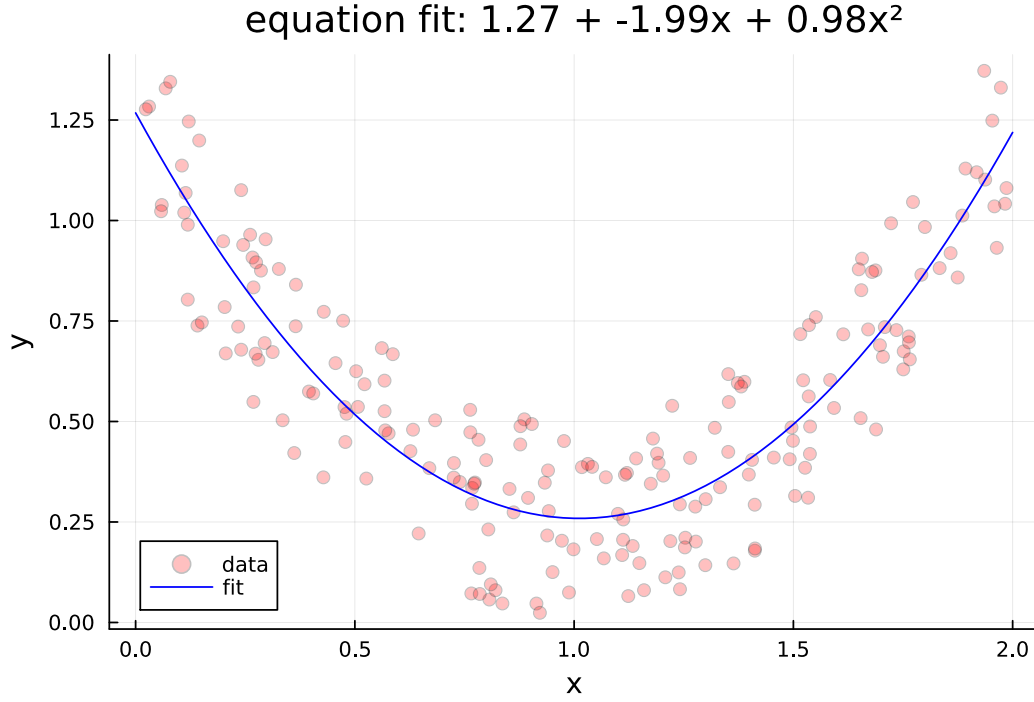


Figure 4.2. An example of polynomial regression for which we fit a quadratic polynomial to some noisy data.

The following example shows how to use such a mapping to produce a quadratic polynomial fit.

We see from Figure 4.2 that our linear regression model found a great fit for a 2nd order polynomial when supplied with polynomial features. Let's update our Bayesian regression scheme to reflect the use of our feature projection map ϕ . First we define

$$\Phi := \phi(X) \tag{4.67}$$

$$\phi_* := \phi(\mathbf{x}_*) \tag{4.68}$$

Our predictive distribution therefore becomes

$$p(y_* | \mathbf{x}_*, X, \mathbf{y}) = \mathcal{N} \left(\frac{1}{\sigma_n^2} \phi_*^T A^{-1} \Phi \mathbf{y}, \phi_*^T A^{-1} \phi_* \right) \tag{4.69}$$

$$A = \frac{1}{\sigma_n^2} \Phi \Phi^T + \Sigma_p^{-1} \tag{4.70}$$

Great! Now we can do our Bayesian inference with non-linear features given by ϕ . There is a *massive* problem with this approach, however. Our order 2 polynomial map ϕ takes us from a D dimensional feature vector to $(D + 1)!$ many. This means that as we add more features to our feature map, the dimension of the resulting vector will quickly become prohibitively large. Looking at our current model equations, we see that the bottleneck is the matrix inversion of A which requires we invert an $N \times N$ matrix. Our prediction (i.e. the mean) involves multiplication on the right by the n dimensional vector \mathbf{y} . With that in mind, perhaps we can reformulate the above into an equivalent form using at most an $n \times n$ dimensional matrix.

Let $K := \Phi^T \Sigma_p \Phi$. Observe the following:

$$\frac{1}{\sigma_n^2} \Phi (K + \sigma_n^2 I) = \frac{1}{\sigma_n^2} \Phi (\Phi^T \Sigma_p \Phi + \sigma_n^2 I) \quad (4.71)$$

$$= \frac{1}{\sigma_n^2} \Phi \Phi^T \Sigma_p \Phi + \Phi I \quad (4.72)$$

$$= \left(\frac{1}{\sigma_n^2} \Phi \Phi^T \right) \Sigma_p \Phi + (\Phi I \Phi^{-1} \Sigma_p^{-1}) \Sigma_p \Phi \quad (4.73)$$

$$= \left(\frac{1}{\sigma_n^2} \Phi \Phi^T + \Sigma_p^{-1} \right) \Sigma_p \Phi \quad (4.74)$$

$$= A \Sigma_p \Phi \quad (4.75)$$

From there we see that

$$A^{-1} \frac{1}{\sigma_n^2} \Phi (K + \sigma_n^2 I) = \Sigma_p \Phi \quad (4.76)$$

$$\Rightarrow \frac{1}{\sigma_n^2} A^{-1} \Phi = \Sigma_p \Phi (K + \sigma_n^2 I)^{-1} \quad (4.77)$$

$$\Rightarrow \frac{1}{\sigma_n^2} \phi_*^T A^{-1} \Phi = \phi_*^T \Sigma_p \Phi (K + \sigma_n^2 I)^{-1} \quad (4.78)$$

For the covariance, we utilize the matrix inversion lemma [\(ADD REFERENCE HERE\)](#) which states

$$(Z + U W V^T)^{-1} = Z^{-1} - Z^{-1} U (W^{-1} + V^T Z^{-1} U)^{-1} V^T Z^{-1} \quad (4.79)$$

With the identification

$$Z^{-1} \rightarrow \Sigma_p \quad (4.80)$$

$$W^{-1} \rightarrow \sigma_n^2 I \quad (4.81)$$

$$V \rightarrow \Phi \quad (4.82)$$

$$U \rightarrow \Phi \quad (4.83)$$

we find

$$\Sigma_p - \Sigma_p \Phi (\Sigma_p + \Phi^T \Sigma_p \Phi)^{-1} \Phi^T \Sigma_p = \left(\Sigma_p^{-1} + \Phi \frac{1}{\sigma_n^2} I \Phi^T \right)^{-1} \quad (4.84)$$

$$= \left(\frac{1}{\sigma_n^2} \Phi \Phi^T + \Sigma_p^{-1} \right)^{-1} \quad (4.85)$$

$$= A^{-1} \quad (4.86)$$

Thus, we have the equivalent form for our predictive distribution:

$$\boxed{p(y_* | \mathbf{x}_*, X, \mathbf{y}) = \mathcal{N} \left(\phi_*^T \Sigma_p \Phi (K + \sigma_n^2 I)^{-1} \mathbf{y}, \phi_*^T \Sigma_p \phi_* - \phi_*^T \Sigma_p \Phi (K + \sigma_n^2 I)^{-1} \Phi^T \Sigma_p \phi_* \right)} \quad (4.87)$$

where the pesky $N \times N$ term has been replaced by the $n \times n$ matrix $\Phi^T \Sigma_p \Phi$.

We now make the *key* observation that the only matrices that appear in the above expression are

$$\Phi^T \Sigma_p \Phi, \quad \phi_*^T \Sigma_p \phi_* \quad (4.88)$$

$$\phi_*^T \Sigma_p \Phi, \quad \Phi^T \Sigma_p \phi_* \quad (4.89)$$

whose matrix elements we can write abstractly as

$$\phi(\mathbf{x})^T \Sigma_p \phi(\mathbf{x}') \quad (4.90)$$

To fit our model, we must determine appropriate values for the symmetric, positive semi-definite covariance matrix Σ_p (and σ_n too, technically). Instead, we observe that this matrix

product is a quadratic form which we can think of as representing an inner product on our transformed vectors:

$$K_{ij} = k(\mathbf{x}_i, \mathbf{x}_j) = \langle \phi(\mathbf{x}_i), \phi(\mathbf{x}_j) \rangle \quad (4.91)$$

We call the function $k(\mathbf{x}, \mathbf{x}')$ the *kernel function* or the *covariance function*.

All we need to perform the above calculations are the matrix elements of K applied to our data \mathcal{D} and any test points \mathbf{x}_* we wish to apply our model to. In effect, this means we are free to use feature vectors of any dimension, including ∞ . The idea here is that the kernel function represents an inner product over *some* vector space. As it turns out, the RBF kernel corresponds to a an infinite dimensional feature vector.

There are many choices for the kernel function. One of the most popular is the RBF (radial basis function) kernel, also commonly referred to as the *squared exponential kernel*:

$$k_{\text{rbf}}(\mathbf{x}, \mathbf{x}') := \sigma_f^2 \exp\left(-\frac{1}{2\ell^2} |\mathbf{x} - \mathbf{x}'|^2\right) \quad (4.92)$$

where σ_f^2 is the *signal variance* and ℓ denotes the similarity length scale.

For notational convenience, let's define

$$K := k(X, X) \quad (4.93)$$

$$K_{**} := k(X_*, X_*) \quad (4.94)$$

$$K_* := k(X, X_*) \quad (4.95)$$

then, our predictive distribution takes the final, *clean* form

$$\boxed{p(\mathbf{y}_* | X_*, X, \mathbf{y}) = \mathcal{N}\left(K_*^T (K + \sigma_n^2 I)^{-1} \mathbf{y}, K_{**} - K_*^T (K + \sigma_n^2 I)^{-1} K_*\right)} \quad (4.96)$$

This is the *end-result* of Gaussian Process Regression acheived via the *weight-space view*.

The Function-space View

So far our approach has been to generalize the standard linear regression model to allow for fitting over a (possibly infinite) basis of features with consideration for measurement and model uncertainty (our Bayesian priors). In essence, the idea was to fit the distribution of all possible weights conditioned on the available training data, $p(\mathbf{w}|X, \mathbf{y})$. A second *equivalent* approach is to instead consider the distribution of all possible model function $f(\mathbf{x})$. By constructing a Bayesian prior over this space, we constrain the space of possible model functions and learn a *distribution* over all allowed model functions, $p(f|X, \mathbf{y})$. To do so we will need to develop the abstract machinery of distributions over function spaces. When these distributions are Gaussian, the result is called a **Gaussian process**.

By this point, we are very familiar with the Gaussian distribution, a.k.a. the Normal distribution $\mathcal{N}(\mu, \sigma^2)$. This distribution is defined by a mean value μ and a variance σ^2 . It's *big brother* is the **Multivariate Normal Distribution**, $\mathcal{N}(\mu, \Sigma)$, described by a vector of means μ and a covariance matrix Σ . A natural question, then, is can we generalize the concept of the Gaussian distribution from N dimensions to being defined over a continuous field? This leads us naturally to the development of a so-called **Gaussian Process**.

Definition: A *Gaussian Process*, \mathcal{GP} , is a collection of random variables for which any finite subset are described by a joint Gaussian distribution.

To see where this comes from, recall that in our previous derivation, we already made the assumption that all our data points \mathcal{D} are i.i.d. Gaussian distributed. A gaussian process is the natural extension of this and makes the assumption that the continuous set from which are data are sampled are *so Gaussian* that any finite sample will be jointly Gaussian distributed. The term *process* is used to distinguish between finite collections of random variables (distributions) and their continuous counterparts described here.

Because each finite subset of this continuous collection is jointly gaussian, we can completely specify a Gaussian Process with two functions: the mean function $m(\mathbf{x})$ and the covariance function $k(\mathbf{x}, \mathbf{x}')$. To denote this, we typically write

$$f(\mathbf{x}) \sim \mathcal{GP}(m(\mathbf{x}), k(\mathbf{x}, \mathbf{x}')) \quad (4.97)$$

To see this in action, recall our Bayesian regression model

$$f(\mathbf{x}) = \phi(\mathbf{x})^T \mathbf{w} \quad \mathbf{w} \sim \mathcal{N}(\mathbf{0}, \Sigma_p) \quad (4.98)$$

where we have set the prior on \mathbf{w} to have zero mean. The mean function is given by the expectation value of our model:

$$\mathbb{E}[f(\mathbf{x})] = \phi(\mathbf{x})^T \mathbb{E}[\mathbf{w}] = 0 \quad (4.99)$$

and the covariance function is given by

$$\mathbb{E}[f(\mathbf{x})f(\mathbf{x}')] = \phi(\mathbf{x})^T \mathbb{E}[\mathbf{w}\mathbf{w}^T] \phi(\mathbf{x}') = \phi(\mathbf{x})^T \Sigma_p \phi(\mathbf{x}') \quad (4.100)$$

To repeat the point, the key feature of Gaussian processes is that finite subsets are jointly Gaussian distributed. Thus we can split our data into the testpoints $\mathcal{D} = (X, \mathbf{y})$ and testpoints X_* and treat each collection as joint distributions with the following priors:

$$\begin{bmatrix} \mathbf{f} \\ \mathbf{f}_* \end{bmatrix} \sim \mathcal{N} \left(\mathbf{0}, \begin{bmatrix} K(X, X) & K(X, X_*) \\ K(X_*, X) & K(X_*, X_*) \end{bmatrix} \right) \quad (4.101)$$

where $\mathbf{f} := f(X)$ and $\mathbf{f}_* = f(X_*)$.

To obtain our predictive distribution, $p(\mathbf{f}_* | X_*, X, \mathbf{y})$, we *condition the joint prior distribution* on the observations. To see how this works, consider a general joint gaussian distribution given by

$$\begin{bmatrix} x \\ y \end{bmatrix} \sim \mathcal{N} \left(\begin{bmatrix} \mu_x \\ \mu_y \end{bmatrix}, \begin{bmatrix} \Sigma_{xx} & \Sigma_{xy} \\ \Sigma_{yx} & \Sigma_{yy} \end{bmatrix} \right) \quad (4.102)$$

define the centered values $\tilde{x} := x - \mu_x$ and $\tilde{y} := y - \mu_y$. Define the intermediate variable

$$z := \tilde{x} - A\tilde{y} \quad (4.103)$$

Note that since we've subtracted out the mean we have $\mathbb{E}[\tilde{x}] = \mathbb{E}[\tilde{y}] = \mathbb{E}[z] = 0$. Let's now find A .

$$\mathbb{E}[z\tilde{y}^T] = \mathbb{E}[(\tilde{x} - A\tilde{y})\tilde{y}^T] \quad (4.104)$$

$$= \mathbb{E}[\tilde{x}\tilde{y}^T - A\tilde{y}\tilde{y}^T] \quad (4.105)$$

$$= \mathbb{E}[\tilde{x}\tilde{y}^T] - \mathbb{E}[A\tilde{y}\tilde{y}^T] \quad (4.106)$$

$$= \Sigma_{xy} - A\mathbb{E}[\tilde{y}\tilde{y}^T] \quad (4.107)$$

$$= \Sigma_{xy} - A\Sigma_{yy} \quad (4.108)$$

Therefore if we choose A so that z and \tilde{y} are independent and uncorrelated, then $\Sigma_{zy} = \mathbb{E}[z\tilde{y}^T] = 0$. Using this assumption, we find

$$0 = \mathbb{E}[z\tilde{y}^T] = \Sigma_{xy} - A\Sigma_{yy} \Rightarrow \boxed{A = \Sigma_{xy}\Sigma_{yy}^{-1}} \quad (4.109)$$

If we now condition \tilde{x} on \tilde{y} (i.e. look at \tilde{x} when \tilde{y} is constant), we find

$$\mathbb{E}[\tilde{x}|\tilde{y}] = A\tilde{y} + \mathbb{E}[z] \quad (4.110)$$

$$= A\tilde{y} + 0 \quad (4.111)$$

$$= \Sigma_{xy}\Sigma_{yy}^{-1}\tilde{y} \quad (4.112)$$

$$(4.113)$$

By manipulating this expression, we can now derive $\mathbb{E}[x|y]$ as follows:

$$\mathbb{E}[x|\tilde{y}] = \mathbb{E}[\tilde{x}|\tilde{y}] + \mu_x \quad (4.114)$$

$$= \mu_x + \Sigma_{xy}\Sigma_{yy}^{-1}\tilde{y} \quad (4.115)$$

$$(4.116)$$

$$\boxed{\mathbb{E}[x|y] = \mu_x + \Sigma_{xy}\Sigma_{yy}^{-1}(y - \mu_y)} \quad (4.117)$$

Similarly for the covariance, we have

$$\text{Cov}(x|y) = \text{Cov}(\tilde{x} + \mu_x|\tilde{y}) \quad (4.118)$$

$$= \text{Cov}(\tilde{x} + \mu_x|\tilde{y} + \mu_y) \quad (4.119)$$

$$= \text{Cov}(\tilde{x}|(\tilde{y} + \mu_y)) \quad (4.120)$$

$$= \text{Cov}(\tilde{x}|\tilde{y}) \quad (4.121)$$

$$= \text{Cov}((z + A\tilde{y})|\tilde{y}) \quad (4.122)$$

$$= \text{Cov}(z) + A\text{Cov}(\tilde{y}) \quad (4.123)$$

$$= \text{Cov}(z) + 0 \quad (4.124)$$

$$= \mathbb{E}[zz^T] \quad (4.125)$$

$$= \mathbb{E}[(\tilde{x} - A\tilde{y})(\tilde{x} - A\tilde{y})^T] \quad (4.126)$$

$$= \mathbb{E}[\tilde{x}\tilde{x}^T - A\tilde{y}\tilde{x}^T - x(A\tilde{y})^T + A\tilde{y}\tilde{y}^T A^T] \quad (4.127)$$

$$= \Sigma_{xx} - A\Sigma_{yx} - \Sigma_{xy}A^T + A\Sigma_{yy}A^T \quad (4.128)$$

$$= \Sigma_{xx} - (\Sigma_{xy}\Sigma_{yy}^{-1})\Sigma_{yx} - \Sigma_{xy}(\Sigma_{yy}^{-1})^T\Sigma_{xy}^T + \Sigma_{xy}\Sigma_{yy}^{-1}\Sigma_y(\Sigma_y^{-1})^T\Sigma_{xy}^T \quad (4.129)$$

$$= \Sigma_{xx} - \Sigma_{xy}\Sigma_{yy}^{-1}\Sigma_{xy}^T - \Sigma_{xy}(\Sigma_{yy}^{-1})^T\Sigma_{xy}^T + \Sigma_{xy}(\Sigma_{yy}^{-1})^T\Sigma_{xy}^T \quad (4.130)$$

$$= \Sigma_{xx} - \Sigma_{xy}[\Sigma_{yy}^{-1} - (\Sigma_{yy}^{-1})^T + (\Sigma_{yy}^{-1})^T]\Sigma_{xy}^T \quad (4.131)$$

$$= \Sigma_{xx} - \Sigma_{xy}\Sigma_{yy}^{-1}\Sigma_{yx} \quad (4.132)$$

$$\boxed{\text{Cov}(x|y) = \Sigma_{xx} - \Sigma_{xy}\Sigma_{yy}^{-1}\Sigma_{yx}} \quad (4.133)$$

Armed with this identity for joint Guassian distributions, we are ready to derive the predictive distribution for Gaussian Process Regression. We find:

$$p(\mathbf{f}_*|X_*, X, \mathbf{y}) = \mathcal{N}(K_*^T K^{-1}\mathbf{f}, K_{**} - K_*^T K^{-1}K_*). \quad (4.134)$$

To account for noisy observations, we can augment our correlation function to include a noise offset. The joint distribution then becomes:

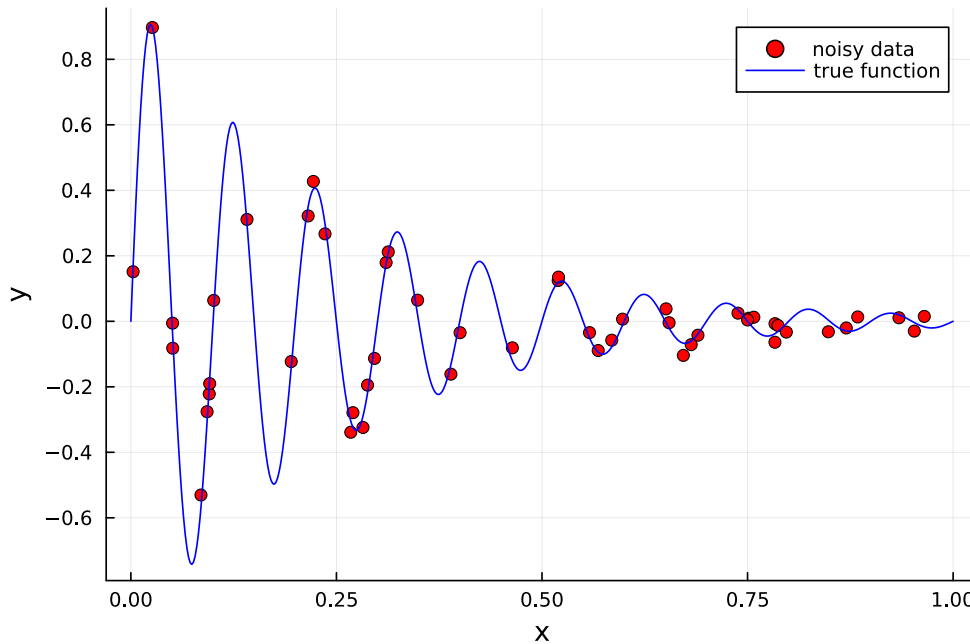
$$\begin{bmatrix} \mathbf{f} \\ \mathbf{f}_* \end{bmatrix} \sim \mathcal{N} \left(\mathbf{0}, \begin{bmatrix} K(X, X) + \sigma_n^2 I & K(X, X_*) \\ K(X_*, X) & K(X_*, X_*) \end{bmatrix} \right) \quad (4.135)$$

which leads to the predictive distribution

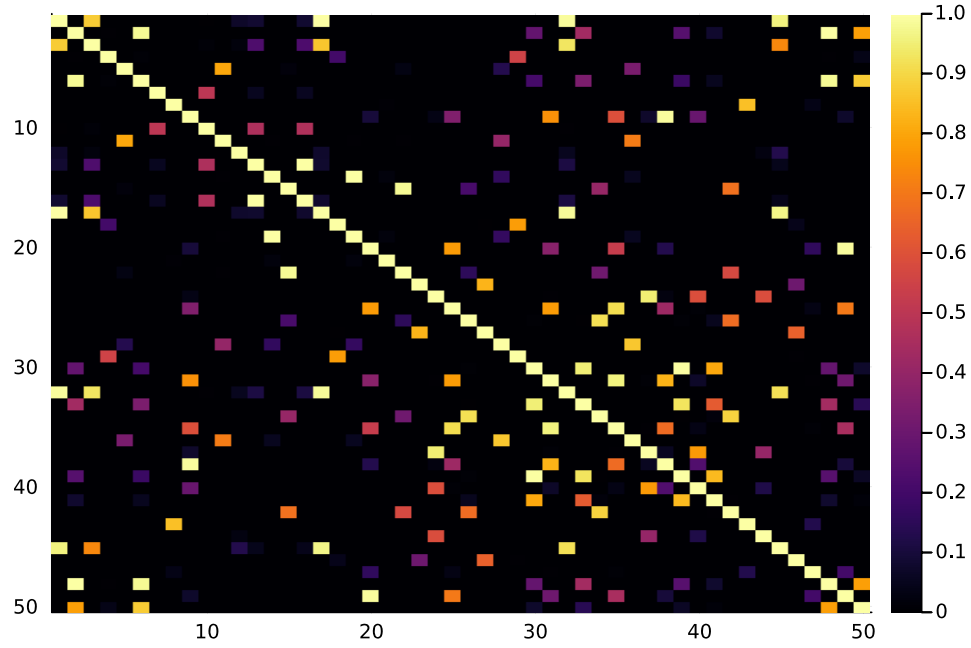
$$p(\mathbf{f}_* | X_*, X, \mathbf{y}) = \mathcal{N} \left(K_*^T [K + \sigma_n^2 I]^{-1} \mathbf{f}, K_{**} - K_*^T [K + \sigma_n^2 I]^{-1} K_* \right) \quad (4.136)$$

Doing it in Julia

To see how we can use this in practice, let's first construct some sample data which we seek to model as a Gaussian Process



The excellent package `KernelFunctions.jl` provides a clean interface to create various kernel functions and apply them to data to create our K -matrices. Due to the fact that kernel functions obey composition laws, we can easily build up complicated Kernels from basic pieces via function composition with `∘`



Unsurprisingly, there is a lot of activation on the diagonal as for a single datapoint \mathbf{x} , we have

$$k(\mathbf{x}, \mathbf{x}) = \exp\left(-\frac{0}{2\ell^2}\right) = 1.0 \quad (4.137)$$

The package `AbstractGPs.jl` provides an excellent way to define Gaussian Processes by supplying mean and kernel functions. We can then sample from our GPs with a simple interface designed to extend the basic functions from `Statistics.jl`. From an `AbstractGP` we can construct a `FiniteGP` by *indexing* into our datasets. The procedure can be summarized as follows

1. Build a kernel function $k(\cdot, \cdot)$ via composition using `KernelFunctions.jl`
2. Construct an a Gaussian Process $f \sim \mathcal{GP}$ abstractly using `AbstractGPs.jl`
3. Construct a finite representation of our GP, f_x , over our training data
4. Construct a posterior Gaussian Process from f_x and our training targets \mathbf{y} .
5. Construct a finite representation of the posterior GP applied to our prediction data

6. Sample this final distribution to obtain a prediction via `mean()` and variances via `var()`. Alternatively, we can obtain a multivariate normal distribution for each point by calling `marginals()`.

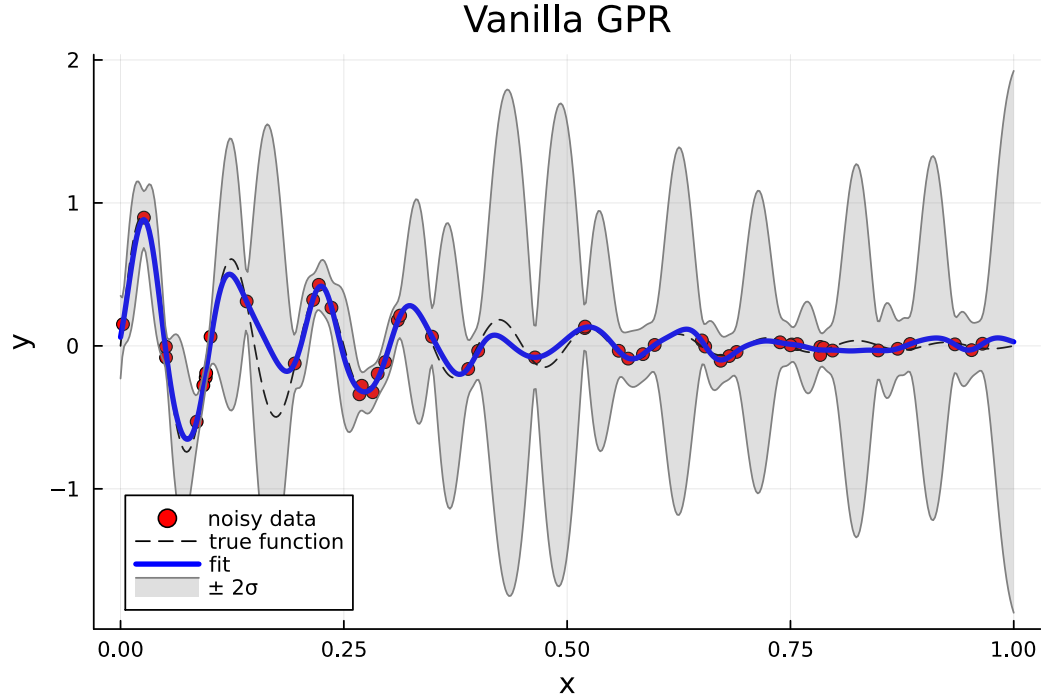


Figure 4.3. A Gaussian Process fit to the training data illustrating the prediction (means) and a $\pm 2\sigma$ uncertainty interval. We can clearly see how the uncertainty is larger for inputs far away from supplied training data.

Fitting the Kernel Hyperparameters

At this point, it is easy to think we are finished; we *already* fit the Gaussian process. However, we were forced to choose values for both ℓ and σ^2 kernel parameters. How can we optimally select the ideal hyperparameters for our Gaussian Process? This question leads us into the realm of Bayesian Model Selection. Rather than focusing specifically on our Gaussian Process model, let's take a step back and think about the process of model selection from a Bayesian perspective.

There are several levels of parameters in machine learning. At the lowest level, we have the model weights \mathbf{w} . Above that, we have model hyperparameters, θ . At the top we have model structure \mathcal{H} . In our Bayesian framework, we can consider prior distributions defined at each of these levels which codify credences for how much we trust a particular model, hyperparameter, etc. At the bottom, we have

$$p(\mathbf{w}|X, \mathbf{y}, \theta, \mathcal{H}_i) = \frac{p(\mathbf{y}|X, \mathbf{w}, \theta, \mathcal{H}_i)p(\mathbf{w}|\theta, \mathcal{H}_i)}{p(\mathbf{y}|X, \theta, \mathcal{H}_i)} \quad (4.138)$$

If this looks confusing, consider Bayes rule for 3 events R, H, S . We have:

$$P(R|H, S) = \frac{P(R, H, S)}{P(H, S)} \quad (4.139)$$

$$= \frac{P(H|R, S)P(R, S)}{P(H, S)} \quad (4.140)$$

$$= \frac{P(H|R, S)P(R|S)P(S)}{P(H|S)P(S)} \quad (4.141)$$

$$= \frac{P(H|R, S)P(R|S)}{P(H|S)} \quad (4.142)$$

To get the result, just think of θ and \mathcal{H}_i as a single *event* and translate the above to distribution functions. See [stack exchange link](#).

The prior $p(\mathbf{w}|\theta, \mathcal{H}_i)$ encodes any knowledge we have about the parameters prior to seeing the data. The denominator is the *marginal likelihood* and is given by

$$p(\mathbf{y}|X, \theta, \mathcal{H}_i) = \int d\mathbf{w} p(\mathbf{y}|X, \mathbf{w}, \theta, \mathcal{H}_i)p(\mathbf{w}|\theta, \mathcal{H}_i) \quad (4.143)$$

The next level up is to express the distribution of hyper-parameters θ :

$$p(\theta|X, \mathbf{y}, \mathcal{H}_i) = \frac{p(\mathbf{y}|X, \theta, \mathcal{H}_i)p(\theta|\mathcal{H}_i)}{p(\mathbf{y}|X, \mathcal{H}_i)}. \quad (4.144)$$

Here $p(\theta|\mathcal{H}_i)$ is called the *hyper-prior*. Similarly, the normalization constant is given by

$$p(\mathbf{y}|X, \mathcal{H}_i) = \int d\theta p(\mathbf{y}|X, \theta, \mathcal{H}_i)p(\theta|\mathcal{H}_i). \quad (4.145)$$

Finally, at the top level we have the set of possible model structures $\{\mathcal{H}_i\}$. This leads to

$$p(\mathcal{H}_i|X, \mathbf{y}) = \frac{p(\mathbf{y}|X, \mathcal{H}_i)p(\mathcal{H}_i)}{p(\mathbf{y}|X)} \quad (4.146)$$

with normlization constant

$$p(\mathbf{y}|X) = \sum_i p(\mathbf{y}|X, \mathcal{H}_i)p(\mathcal{H}_i). \quad (4.147)$$

Depending on the model details, these integrals may be intractable without approximations or Monte Carlo methods. Since we rarely have sufficient knowledge to form a hyperparameter prior, one often attempts to maximize the marginal likelihood $p(\mathbf{y}|X, \theta, \mathcal{H}_i)$ with respect to the hyperparameters θ instead. This is known as Type II Maximum Likelihood Estimation. (Refer to prior section on this in the Theoretical Techniques chapter). In the case of Gaussian Process Regression, we are once again saved by the fact that every piece has a convenient functional form resulting in analytically tractable integrals for the marginal likelihood function. We find

$$\ln p(\mathbf{y}|X, \theta) = -\frac{1}{2}\mathbf{y}^T(K_f + \sigma_n^2 I)^{-1}\mathbf{y} - \frac{1}{2}\ln|K_f + \sigma_n^2 I| - \frac{n}{2}\ln(2\pi) \quad (4.148)$$

where we have employed the natural logarithm to remove the pesky exponentials and arrive at a very convenient form. Note that because the logarithm is monotonically increasing, the maximum of the log-marginal-likelihood will be the same as the marginal-likelihood itself. Provided this functional form, we can use our favorite optimization routine to obtain the kernel hyperparameters which maximize the likelihood of obtaining our data as illustrated in the following comparison figure.

4.4.3 Decision Trees

4.5 Unsupervised Classification

4.5.1 Self Organizing Maps

Self organizing maps (SOMs) are an unsupervised machine learning technique developed by Kohonen (Kohonen, 1982) based on the simple biological principle that *neurons near each*

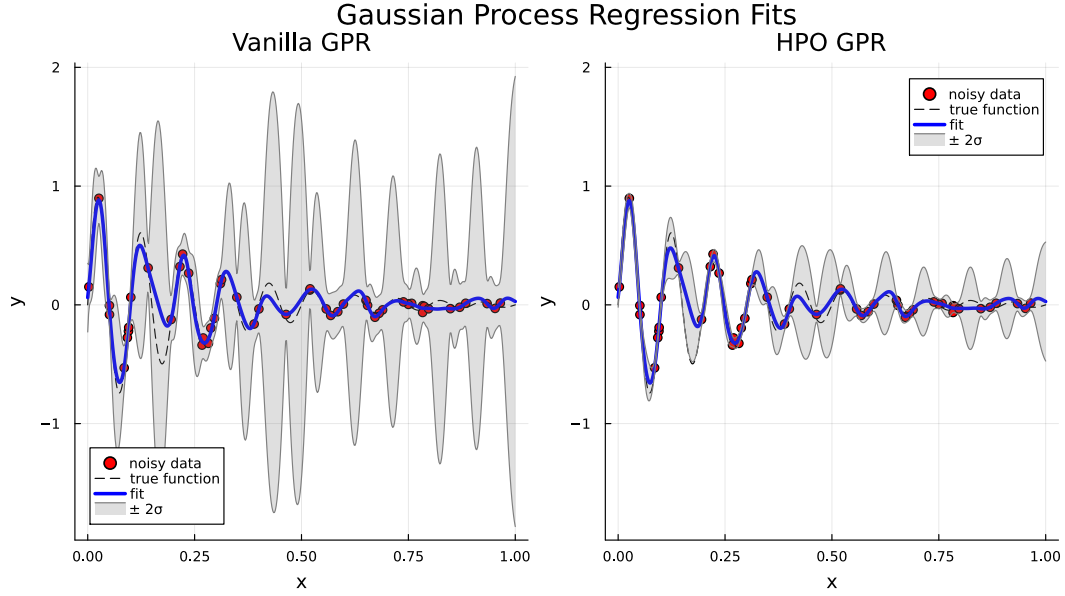


Figure 4.4. Left: the original Gaussian Process obtained with our default choice of kernel parameters. Right: A much better Gaussian Process obtained after performing hyperparameter optimization on the kernel parameters. Note how the mean function still does an excellent job fitting the data points while *also* minimizing the uncertainty of the fit.

other fire together. This observation that the topological *closeness* of similar computational units is a critical feature of intelligent systems leads to a natural reinterpretation of the familiar perceptron model into a new form amenable for a variety of clustering and dimensionality reduction tasks. In particular, the SOM enables a rapid unsupervised classification of multidimensional data into a (typically) one or two dimensional *simplicial complex*, the discrete realization of a topological manifold, whose vertices correspond to representative points in the original data space \mathcal{D} . While a tad esoteric compared to other popular unsupervised methods like KMeans clustering or DBSCAN, the SOM distinguishes itself with the added benefit that it's training procedure guarantees nodes (i.e. classes) close to each other in the feature manifold share similar weights. This additional structure makes the SOM particularly attractive when an interpretation of the discovered clusters as well as the relationships between them is desired.

The original treatment of the SOM by Kohonen was made in terms of processing units, sensory signals, and relaying networks (Kohonen, 1982), however, in the modern era of deep learning, a more easily digestible derivation can be obtained by re-interpreting the weights of a simple perceptron model to provide the foundation for a clustering approach. As described in previously, a perceptron is a function of the form

$$\mathbf{y} = \sigma.(W\mathbf{x}) \quad (4.149)$$

where $W \in \mathbb{R}^{n \times m}$ is a matrix of weights which transform the input $\mathbf{x} \in \mathbb{R}^m$ into \mathbb{R}^n and σ is a nonlinear *activation function* applied element-wise to the outputs of the matrix multiplication (indicated by the $.$ syntax). If we instead think of the weight matrix as an ordered collection of vectors $\{\mathbf{w}_i\}_{i=1}^n$, then this formula can be further decomposed into

$$(\mathbf{y})_i = \sigma(\mathbf{w}_i^T \mathbf{x}) = \sigma(\langle \mathbf{w}_i, \mathbf{x} \rangle) \quad (4.150)$$

The function of the perceptron is now clear: given an input vector \mathbf{x} and a collection of n -many weight vectors \mathbf{w}_i , compute the n -many inner products of \mathbf{x} with each weight vector \mathbf{w}_i , apply the nonlinear activation function σ , and concatenate the results.

If we now allow ourselves to imagine the weight vectors \mathbf{w}_i as members of the same vector space as the input \mathbf{x} , a reasonable question to ask is: *how similar is the input \mathbf{x} to each \mathbf{w}_i .* Further, the application of the inner product $\langle \cdot, \cdot \rangle$ suggests we may answer this question in terms of the distance

$$\langle \mathbf{w}_i - \mathbf{x}, \mathbf{w}_i - \mathbf{x} \rangle = d(\mathbf{w}_i, \mathbf{x})^2. \quad (4.151)$$

In other words, given a set of weight vectors \mathbf{w}_i which we may now think of as the cluster centers for our unsupervised model, we can measure the similarity between a given datum \mathbf{x}_j and each cluster by computing the distance

$$d_{ij} = d(\mathbf{w}_i, \mathbf{x}_j). \quad (4.152)$$

To make this mapping useful, we should prescribe a *training procedure* which updates the vectors \mathbf{w}_i based on all of the available data points. Further, our goal is to establish some kind of interpretable relationship between the various weights \mathbf{w}_i so that they *more than* disjoint classes. The SOM achieves this by providing a lower-dimensional grid (typically 2-dimensional) whose vertices are understood to be the location of each of the weight vectors. The procedure is as follows:

First, initialize a grid of weight vectors $\mathbf{w}_i \in \mathbb{R}^m$ which we collect into a matrix $W \in \mathbb{R}^{m \times n}$. Assign coordinates to each weight vector by mapping each \mathbf{w}_i to a vertex in the grid. The *topology* of the grid is a hyperparameter which dictates the spatial relationship between neighboring nodes. Popular choices are: flat rectangular, rectangular on a cylinder (one periodic boundary condition), rectangular on a torus (two periodic boundary conditions), hexagonal (more equidistant neighbors per node than rectangular), and spherical. One can

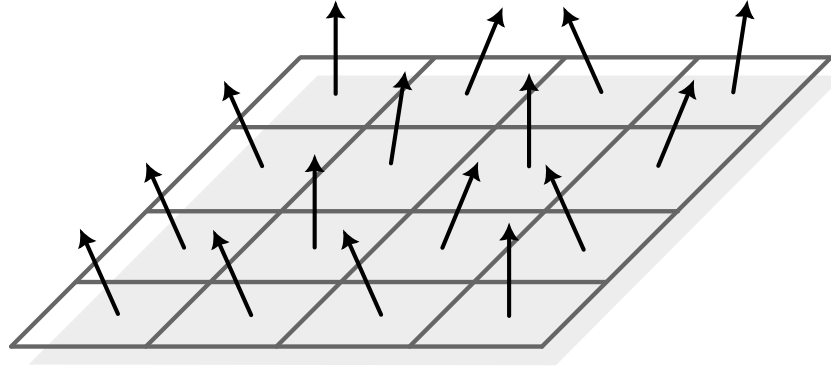


Figure 4.5. A Self Organizing Map with nodes configured in a rectangular grid topology with weight vectors randomly initialized.

initialize the weights in a variety of ways. Some common choices are

- Randomly initialize them as in figure 4.5.

- Initialize them to the value of n -many randomly selected data samples \mathbf{x}_i .
- Initialize them to the first n -many principal components of the dataset.

Next, we proceed to update the weights according to the following steps for each \mathbf{x}_k datum:

1. Compute the *best match unit* (BMU) as

$$\mathbf{w}_i^{\text{best}} = \arg \min_{\mathbf{w}_j} (d(\mathbf{x}_k, \mathbf{w}_j)^2) \quad (4.153)$$

where $d(\cdot, \cdot)$ is some suitably chosen distance function. One often defaults to the Euclidean metric.

2. Choose a radius σ_t which we will use to identify the correct update for to apply to all nodes relative to $\mathbf{w}_i^{\text{best}}$.
3. Update *all* the weights according to the equation

$$w_m^{t+1} = w_m^t + \eta(t) f(x_m, y_m, \sigma_t) (\mathbf{x}_k - \mathbf{w}_n^t) \quad (4.154)$$

where $\eta(t)$ is the learning rate, and $f(x_m, y_m, \sigma_t)$ is the *neighborhood* function which defines the relative amount each node \mathbf{w}_m is updated according to its coordinate distance from the best match unit (i.e. the distance between the points (x_m, y_m) and $(x_{\text{best}}, y_{\text{best}})$ in the two dimensional case). Typically one chooses a the learning rate to evolve according to the schedule

$$\eta(t) = \eta_0 \exp(-\lambda t) \quad (4.155)$$

with the radius σ_t evolving similarly according to

$$\sigma_t = \sigma_0 \exp(-\beta t). \quad (4.156)$$

Popular choices for the neighborhood function are the Gaussian function, Mexican-hat function, a cone function, and a cylinder function.

A simple example: partitioning color spaces

As an illustrative example, consider the minimal dataset formed by the colors red, green, and blue which as vectors may be written as

$$\text{Red} = (1, 0, 0) \tag{4.157}$$

$$\text{Green} = (0, 1, 0) \tag{4.158}$$

$$\text{Blue} = (0, 0, 1). \tag{4.159}$$

Now let us suppose we want to train an SOM to generate a classification for colors using only these three as the supplied training data. Figure 4.6 shows exactly this. Here we have trained an SOM of size 25×25 cells in a flat hexagonal topology. As the weight vectors are themselves vectors of length 3, we can then visualize the trained SOM by coloring each corresponding node by the color learned in its weight vector. What we find is that the resulting map has clearly separated red, green, and blue into distinct regions in the grid, and more importantly, there is a smooth gradient between neighboring cells reflecting the fact that neighboring classes are more *similar* than cells with a large separation distance. This is not the case for many other common clustering techniques like k-nearest neighbors or DBSCAN for which relationship between learned classes is challenging to interpret.

To enable this demonstration and its subsequent use throughout the rest of this dissertation, I have created a new open-source implementation of the SOM algorithm for the Julia programming language called `SelfOrganizingMaps.jl` which has been added to the general registry and can be freely downloaded for use by anyone. The repository for the code as well as the associated documentation can be found at this site.

4.5.2 Generative Topographic Maps

Now that we've developed the SOM algorithm, we see upon reflection that there are a few drawbacks to the methods, namely

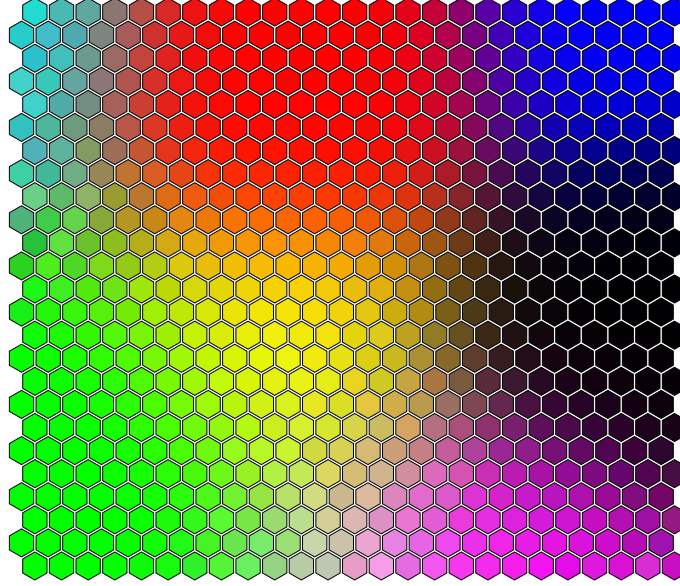


Figure 4.6. A trained Self Organizing Map of 25×25 cells in a hexagonal topology trained on a dataset consisting of the three colors red, green, and blue.

- There is no probabilistic interpretation of the fitted SOM? Are we to trust the fit results with 100% certainty?
- There is no clearly stopping criteria for the learning process? How many epochs should be performed? Do you simply stop the training once the neighbor radius σ_t is less than the minimum neighbor distance?

With this in mind, we now seek to introduce a new method based on the SOM which will allow us to address the shortcomings by taking a principled, probabilistic approach, which will be called the *Generative Topographic Mapping* (GTM). This technique was first introduced by Bishop et al. in (Bishop et al., 1998) and our derivation of the method will follow their presentation closely.

Both the SOM and GTM rely on the assumption that our high dimensional dataset $\mathcal{D} \subseteq \mathbb{R}^D$ can be accurately represented as being constrained to a lower dimensional submanifold. In the SOM this is achieved by assigning coordinates to each of the weight vectors

according to some predetermined grid. In the GTM, however, we make a slight change in our perspective and say that the high dimensional data we have are *generated* from some lower dimensional set of *latent vectors* living in \mathbb{R}^L with $L < D$. For ease of visualization, it is standard to chose $L = 2$. Our goal, then, is to learn a mapping from the latent space to the data space which *maximizes the likelihood* of obtaining our dataset given the set of latent variables.

In the coming derivation, we shall assume the following notations:

$$\begin{aligned}
\mathbf{x} \in \mathbb{R}^L & \quad \text{A latent vector} \\
\mathbf{t} \in \mathbb{R}^D & \quad \text{A data vector} \\
\mathbf{y} = \mathbf{y}(\mathbf{x}; W) & \quad \text{The transformation } \mathbf{y} : \mathbb{R}^L \rightarrow \mathbb{R}^D \\
W & \quad \text{The model weights}
\end{aligned} \tag{4.160}$$

To get started, let us assume that our data can be reasonably described by a multivariate normal distrubtion such that

$$p(\mathbf{t}|\mathbf{x}; W, \beta) = \mathcal{N}\left(\mathbf{y}(\mathbf{x}; W), \frac{1}{\beta}\right), \tag{4.161}$$

that is, our data follow a normal distribution with mean given by the transformation function \mathbf{y} applied to the latent variables \mathbf{x} with covariance $\mathbf{1}/\beta$. Suppose that we have some model for the prior distribution of our latent variables, $p(\mathbf{x})$, then by integration would find

$$p(\mathbf{t}|W, \beta) = \int d\mathbf{x} p(\mathbf{t}|\mathbf{x}; W, \beta) p(\mathbf{x}) \tag{4.162}$$

which is the distribution we really care about, namely, the distribution of data $\mathbf{t} \in \mathcal{D}$ given our choice of model weights W and covariance β^{-1} . To make the problem tractible, we need an integrable distribution for $p(\mathbf{x})$. We therefore take inspiration from the SOM and choose \mathbf{x} to be precisely distributed over a regular grid of points with equal weights. That is

$$p(\mathbf{x}) = \frac{1}{K} \sum_k^K \delta(\mathbf{x} - \mathbf{x}_k) \tag{4.163}$$

where \mathbf{x}_k are the K -many grid points such that for any \mathbf{x} , there is a probability of precisely $1/K$ that it *came from* one of the nodes \mathbf{x}_k .

NOTE: Add figure of GTM grid here. (See pg 27 of notebook)

As mathematicians (or physicists, or data scientists, etc...) we rejoice at the wise decision to incorporate the integral-zapping Dirac-delta distributions which enables us to simplify the data distribution to become

$$\begin{aligned} p(\mathbf{t}|W, \beta) &= \int d\mathbf{x} p(\mathbf{t}|\mathbf{x}; W, \beta) \frac{1}{K} \sum_k^K \delta(\mathbf{x} - \mathbf{x}_k) \\ &= \frac{1}{K} \sum_k^K p(\mathbf{t}|\mathbf{x}_k; W, \beta). \end{aligned} \tag{4.164}$$

Now provided we have in supply a dataset with N -many records, i.e. $\mathcal{D}^N \subseteq \mathcal{D} = \{\mathbf{t}_1, \mathbf{t}_2, \dots, \mathbf{t}_N\}$, how do we chose the *optimal* parameters W and β ? We need an objective to optimize!

If we assume the \mathbf{t}_i are independently, identically distributed, then the *likelihood* of obtaining the data given a choice of parameters (i.e. $p(\mathcal{D}^N|W, \beta)$) is

$$\begin{aligned} \mathcal{L}(W, \beta) &= \prod_n^N p(\mathbf{t}_n|W, \beta) \\ &= \prod_n^N \left(\frac{1}{K} \sum_k^K p(\mathbf{t}_n|\mathbf{x}_k; W, \beta) \right). \end{aligned} \tag{4.165}$$

Taking the logarithm (which must share the same optimum as it is a monotonically increasing function) yields the simpler expression

$$\ell(W, \beta) = \log(\mathcal{L}(W, \beta)) = \sum_n^N \log \left(\frac{1}{K} \sum_k^K p(\mathbf{t}_n|\mathbf{x}_k; W, \beta) \right) \tag{4.166}$$

Maximizing this log-likelihood function is known as *maximum likelihood estimation* (MLE). But how should we optimize this function? We could use techniques like gradient descent or ADAM, however the probabilistic nature of the GTM suggests we may be able to obtain an *expectation-maximization* (EM) routine for a suitable choice of transformation $\mathbf{y}(\mathbf{x}, W)$.

Suppose we already have some guesses W_o and β_o for the parameters (o for *old*). To simplify the derivation, we define $\theta_o = (W_o, \beta_o)$. Then we can compute the responsibilities r_{kn} as

$$\begin{aligned}
r_{kn} &:= p(\mathbf{x}_k | \mathbf{t}_n, \theta_o) \\
&= \frac{p(\mathbf{t}_n | \mathbf{x}_k, \theta_o) p(\mathbf{x}_k | \theta_o)}{p(\mathbf{t}_n | \theta_o)} \\
&= \frac{p(\mathbf{t}_n | \mathbf{x}_k, \theta_o) p(\mathbf{x}_k | \theta_o)}{\sum_{k'}^K p(\mathbf{t}_n | \mathbf{x}_{k'}, \theta_o) p(\mathbf{x}_{k'} | \theta_o)} \\
&= \frac{p(\mathbf{t}_n | \mathbf{x}_k, \theta_o) \frac{p(\mathbf{x}_k, \theta_o)}{p(\theta_o)}}{\sum_{k'}^K p(\mathbf{t}_n | \mathbf{x}_{k'}, \theta_o) \frac{p(\mathbf{x}_{k'}, \theta_o)}{p(\theta_o)}} \\
&= \frac{p(\mathbf{t}_n | \mathbf{x}_k, \theta_o) p(\mathbf{x}_k, \theta_o)}{\sum_{k'}^K p(\mathbf{t}_n | \mathbf{x}_{k'}, \theta_o) p(\mathbf{x}_{k'}, \theta_o)} \\
&= \frac{p(\mathbf{t}_n | \mathbf{x}_k, \theta_o) p(\mathbf{x}_k) p(\theta_o)}{\sum_{k'}^K p(\mathbf{t}_n | \mathbf{x}_{k'}, \theta_o) p(\mathbf{x}_{k'}) p(\theta_o)} \\
&= \frac{p(\mathbf{t}_n | \mathbf{x}_k, \theta_o) p(\mathbf{x}_k)}{\sum_{k'}^K p(\mathbf{t}_n | \mathbf{x}_{k'}, \theta_o) p(\mathbf{x}_{k'})} \\
&= \frac{p(\mathbf{t}_n | \mathbf{x}_k, \theta_o) \frac{1}{K}}{\sum_{k'}^K p(\mathbf{t}_n | \mathbf{x}_{k'}, \theta_o) \frac{1}{K}} \\
&= \frac{p(\mathbf{t}_n | \mathbf{x}_k, \theta_o)}{\sum_{k'}^K p(\mathbf{t}_n | \mathbf{x}_{k'}, \theta_o)}
\end{aligned} \tag{4.167}$$

which we will soon utilize in our EM procedure.

Now, we must chose a form for the transformation $\mathbf{y}(\mathbf{x}, W)$ after which we will have all of the information we need to perform the computation. A convenient choice is to use a kernelized regression strategy so that

$$\mathbf{y} := \phi^T(\mathbf{x})W \tag{4.168}$$

where $W \in \mathbb{R}^{M \times D}$ are some constant parameters and $\phi : \mathbb{R}^L \rightarrow \mathbb{R}^M$ is a transformation employing M -many basis functions ϕ_m . To capture linear and nonlinear effects, we chose

$$\phi_m(\mathbf{x}) = \begin{cases} \exp(-\frac{|\mathbf{x} - \mu_m|^2}{2\sigma}), & m < M \\ 1, & m = M \end{cases} \tag{4.169}$$

that is, $M - 1$ Gaussians and a term for a bias offset. If we apply the transformation to each of the K -many latent nodes, then we may collect the resulting vectors into a matrix,

$$Y = \Phi W, \quad (4.170)$$

where $\Phi \in \mathbb{R}^{K \times M}$ with $\Phi_{km} = \phi_m(\mathbf{x}_k)$.

Now that we are able to compute the responsibilities of each latent node to the observed data (the expectation step), we need to compute the relevant derivatives which we will use in the maximization step. For this, we treat the responsibilities r_{kn} as fixed, and solve for the relevant updates to W and β which make there derivatives 0 and therefore optimize ℓ . Let's begin by differentiating ℓ with respect to the weight matrix W . We have

$$0 = \frac{\partial}{\partial W_{md}} \ell(W, \beta) \quad (4.171)$$

$$= \frac{\partial}{\partial W_{md}} \sum_n \log \left(\frac{1}{K} \sum_k p(\mathbf{t}_n | \mathbf{x}_k; W, \beta) \right) \quad (4.172)$$

$$= \sum_n \frac{1}{\frac{1}{K} \sum_{k'} p(\mathbf{t}_n | \mathbf{x}_{k'}; W, \beta)} \frac{1}{K} \frac{\partial}{\partial W_{md}} \sum_k p(\mathbf{t}_n | \mathbf{x}_k; W, \beta) \quad (4.173)$$

Noting that $p(\mathbf{t}_n | \mathbf{x}_k; W, \beta) = \mathcal{N}(\mathbf{y}_k, \beta^{-1})$, then the derivative of the exponential yields

$$0 = \sum_n \sum_k \frac{p(\mathbf{t}_n | \mathbf{x}_k; W, \beta)}{\sum_{k'} p(\mathbf{t}_n | \mathbf{x}_{k'}; W, \beta)} \frac{\partial}{\partial W_{md}} \left(-\frac{\beta}{2} |\mathbf{t}_n - \mathbf{y}_k|^2 \right) \quad (4.174)$$

$$= \sum_n \sum_k r_{kn} \frac{\partial}{\partial W_{md}} \left(-\frac{\beta}{2} |\mathbf{t}_n - \mathbf{y}_k|^2 \right) \quad (4.175)$$

$$= \sum_n \sum_k (-\beta) r_{kn} \sum_q (y_k^q - t_n^q) \frac{\partial}{\partial W_{md}} \sum_s \phi_s(\mathbf{x}_k) W_{sq} \quad (4.176)$$

$$= \sum_n \sum_k \sum_q (-\beta) r_{kn} (y_k^q - t_n^q) \sum_s \phi_s(\mathbf{x}_k) \delta_{sm} \delta_{qd} \quad (4.177)$$

$$= \sum_n \sum_k (-\beta) r_{kn} (y_k^d - t_n^d) \phi_m(\mathbf{x}_k) \quad (4.178)$$

Looking closely at the above expression, we see that there are two free indices suggesting we may write the above as a matrix expression. To do so, let's introduce a diagonal matrix G such that $G_{kk} = \sum_n r_{kn}$. Then upon rearrangement, we find

$$\sum_n \sum_k r_{kn} y_k^d \phi_m(\mathbf{x}_k) = \sum_n \sum_k r_{kn} t_n^d \phi_m(\mathbf{x}_k) \quad (4.179)$$

$$\sum_n \sum_k r_{kn} \left(\sum_s W_{sd} \phi_s(\mathbf{x}_k) \right) \phi_m(\mathbf{x}_k) = \sum_n \sum_k r_{kn} t_n^d \phi_m(\mathbf{x}_k) \quad (4.180)$$

$$\sum_n \sum_k \sum_s r_{kn} \Phi_{ks} W_{sd} \Phi_{km} = \sum_n \sum_k r_{kn} t_n^d \Phi_{km} \quad (4.181)$$

$$\sum_k \sum_s \left(\sum_n r_{kn} \right) \Phi_{ks} W_{sd} \Phi_{km} = \sum_n \sum_k r_{kn} t_n^d \Phi_{km} \quad (4.182)$$

$$\sum_k \sum_s G_{kk} \Phi_{ks} W_{sd} \Phi_{km} = \sum_n \sum_k r_{kn} t_n^d \Phi_{km} \quad (4.183)$$

$$\sum_k \sum_s (\Phi_{km})^T G_{kk} \Phi_{ks} W_{sd} = \sum_n \sum_k (\Phi_{km})^T r_{kn} t_n^d \quad (4.184)$$

$$(\Phi^T G \Phi W)_{md} = (\Phi^T R T)_{md} \quad (4.185)$$

where we have defined $T_{nd} = t_n^d$ and $R_{kn} = r_{kn}$.

By the same procedure, we find that differentiation with respect to β leads to

$$\frac{1}{\beta} = \frac{1}{ND} \sum_n \sum_k r_{kn} |\mathbf{y}_k - \mathbf{t}_n|^2 \quad (4.186)$$

so that together the maximization step amounts to solving

$$\begin{cases} \Phi^T G_{\text{old}} \Phi W_{\text{new}} = \Phi^T R_{\text{old}} T \\ \frac{1}{\beta_{\text{new}}} = \frac{1}{ND} \sum_n \sum_k R_{kn}^{(\text{old})} |\mathbf{y}_k - \mathbf{t}_n|^2 \end{cases} \quad (4.187)$$

The last piece we need for the training algorithm is a method to initialize the parameters W and β . As before with the SOM, there are a few different options. In particular we might

- Randomly initialize the weight matrix W

- Use PCA to initialize the weights to a linear model. To do this, one computes the data covariance matrix U keeping only the first two columns. We then initialize W so that

$$W\Phi^T \approx UX^T \quad (4.188)$$

and then set β^{-1} to the third principal component variance.

In summary, the GTM training procedure is as follows

1. Generate a grid of latent points $\{\mathbf{x}_k\}_{k=1}^K$.
2. Generate basis function centers $\{\mu_m\}_{m=1}^M$.
3. Select an basis function width σ .
4. Compute the matrix of activations Φ where

$$\Phi_{mk} = \Phi_m(\mathbf{x}_k) \quad (4.189)$$

5. Initialize the weight matrix W randomly or with PCA
6. Initialize β randomly or with PCA
7. (Optional) select a value for α to enable weight regularization, i.e. $(\Phi^T G \Phi + \frac{\alpha}{\beta} I)W = \Phi^T R T$. This corresponds to specifying a prior distribution on the weights W such that $p(W) \propto \exp(-\alpha ||W||^2/2)$.
8. Compute the difference matrix $\Delta := |T - \Phi W|^2$
9. Repeat the following until convergence:
 - (a) (Expectation step) Compute the responsibility matrix R using Δ and β .
 - (b) Compute G from R

- (c) (Maximization step) Compute the update to the weight matrix W with $W_{\text{new}} = (\Phi^T G \Phi)^{-1} \Phi^T R T$
- (d) Compute the updated difference matrix Δ
- (e) Compute the updated β

Excellent! We now have a robust procedure to fit the GTM model. With a fitted GTM in hand, how do we utilize the results? The main idea is that the fitted GTM provides us all of the relevant probability distributions to explain how we obtained our current data \mathbf{t}_n from the latent vectors \mathbf{x}_k . Using Baye's rule, we can invert this relationship using our responsibility matrix R to understand the contributions of each latent vector \mathbf{x}_k to each data. Since this would lead to a different R matrix for each datapoint, one often computes a handful of summary statistics using R rather than using the entire result. For example, we find the mean of the distribution to be

$$\begin{aligned}
\langle \mathbf{x} | \mathbf{t}_n; W, \beta \rangle &= \int d\mathbf{x} p(\mathbf{x} | \mathbf{t}_n; W, \beta) \mathbf{x} \\
&= \int d\mathbf{x} \frac{p(\mathbf{t}_n | \mathbf{x}; W, \beta) p(\mathbf{x})}{\int d\xi p(\mathbf{t}_n | \xi; W, \beta) p(\xi)} \mathbf{x} \\
&= \int \sum_k \frac{p(\mathbf{t}_n | \mathbf{x}, W, \beta) \mathbf{x} \delta(\mathbf{x} - \mathbf{x}_k)}{\sum_{k'} p(\mathbf{t}_n | \mathbf{x}_{k'}; W, \beta)} d\mathbf{x} \\
&= \sum_k \frac{p(\mathbf{t}_n | \mathbf{x}_k; W, \beta) \mathbf{x}_k}{\sum_{k'} p(\mathbf{t}_n | \mathbf{x}_{k'}; W, \beta)} \\
&= \sum_k^K R_{kn} \mathbf{x}_k
\end{aligned} \tag{4.190}$$

However, in cases where the distribution is multi-modal, the mean can be misleading. It can therefore also be useful to compute the mode of the distribution which is given by

$$\mathbf{x}_{\text{mode}} = \mathbf{x}_{k_{\text{max}}} \tag{4.191}$$

where $k_{\text{max}} = \arg \max_k (R_{kn})$.

To provide an easy to use open-source implementation of the GTM algorithm, I have created a publicly accessible github repo with a Julia implementation that comports with the MLJ.jl machine learning framework. The repo can be found [here](#) and makes it easy to incorporate GTMs into complicated machine learning pipelines.

UPDATE REQUIRED: Add demo use for GTM like we did for the SOM.

4.6 Model Ensembling

MLJ and SciKitLearn documentation sites will have good references for this, I think.

4.6.1 Bagging

e.g. Random Forests

4.6.2 Boosting

e.g. XGBoost

4.6.3 Stacking

e.g. Super Learners. Use the example of model stacking from MLJ documentation to describe our approach.

4.7 Uncertainty Quantification via Conformal Prediction

Having established a variety of machine learning methods which we will use throughout the rest of this dissertation, a natural next question is: How can we evaluate our confidence in the predictions of machine learning models? For some methods like Gaussian Process Regression, our models naturally output distributions which we can evaluate to provide predictions via the mean, and uncertainty estimates via the standard deviation or some similar statistic.

For models like Neural Networks and Decision Trees which are not inherently probabilistic, there is no obvious way to extract uncertainty estimates purely from the model’s predictions. One standard approach is to attempt to evaluate our confidence in a particular model by comparing the predictions of many copies of the same model trained on complementary cross validation folds of the original training set. By examining the prediction variance due to variation in the supplied training data, we can establish some expectation for uncertainty in our model’s output.

Clearly this approach is far from perfect. Therefore, in order to prevent ourselves from biasing towards using only those methods like Gaussian Process Regression which output distributions by default, we would like to develop a robust procedure for augmenting *any* regression model with the capability to simultaneously estimate target values and *confidence intervals* by taking advantage of our (often) large datasets to establish sufficient statistics. Further, we do not wish to make any assumptions about the particular target distributions; without sufficient reason to expect a variable to be normally distributed why should we make this *strong* assumption? This is the approach taken by *Conformal Prediction* which we will develop in this section and utilize throughout the rest of this dissertation.

As discussed in the previous section, the typical machine learning procedure involves partitioning our dataset into independent training and testing batches (or k -many cross-validation folds) so that we can evaluate a trained model’s performance and ensure we have not run afoul of over/under-fitting. In conformal prediction, we introduce an additional split so that our original dataset \mathcal{D} can be decomposed into $\mathcal{D} = \mathcal{D}_{\text{train}} \cup \mathcal{D}_{\text{cal}} \cup \mathcal{D}_{\text{test}}$. We then start with *any* heuristic notion of model uncertainty, and make the estimated confidence interval rigorous by adjusting the output to *guarantee* the desired coverage is satisfied on the calibration set \mathcal{D}_{cal} .

A high level outline of the procedure is as follows

1. Define a score function $s(x, y) \in \mathbb{R}$ using some heuristic notion of uncertainty so that large s means *worse* agreement between $\hat{f}(x)$ and the target y .

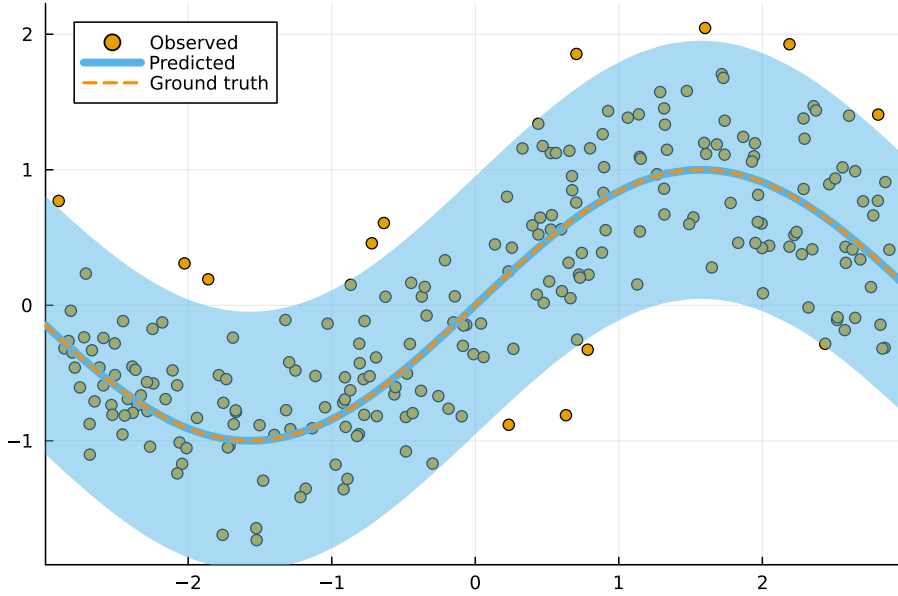


Figure 4.7. An example of conformal prediction for a model estimating a noisy one-dimensional target. The blue shaded region illustrates the learned confidence interval. Image taken from (Altmeyer, 2023)

2. Compute \hat{q} as the $\frac{\lceil (n+1)(1-\alpha) \rceil}{n}$ quantile of the scores $s_i = s(x_i, y_i)$ generated from the calibration set \mathcal{D}_{cal} .
3. Use \hat{q} to form valid prediction intervals.

The question then becomes: How can we obtain accurate uncertainty estimates if the score function s need not be accurate? So long as the scores s_i accurately rank estimates from best to worst model error, then all we need to do is *calibrate* the scores to achieve a confidence interval with the desired coverage, i.e. so that the interval is always large enough to guarantee the coverage we require. It may be that our predicted interval is *too large* for a poor scoring function. As an example, let's demonstrate how we can form correct estimates for the specific problem of quantile regression.

4.7.1 Conformalizing Quantile Regression

In quantile regression the goal is to predict the γ^{th} quantile of the targets y given the input features x . Let us denote $t_\gamma(x)$ as the true value for the γ^{th} quantile and $\hat{t}_\gamma(x)$ as our predicted value. In this framework, we could then expect the interval $[t_{0.05}, t_{0.95}]$ to constitute a 90% coverage confidence interval. However, our fitted values \hat{t}_γ will certainly be imperfect so that the actual interval obtained will not conform to the desired 90%. Suppose we have a holdout dataset \mathcal{D}_{cal} as previously mentioned, and for notational consistency, let α be the coverage so that we may write our predicted interval as $[\hat{t}_{\alpha/2}, \hat{t}_{1-\alpha/2}]$. A reasonable score function (i.e. uncertainty heuristic) would be

$$s(x, y) = \max \{ \hat{t}_{\alpha/2}(x) - y, y - \hat{t}_{1-\alpha/2}(x) \} \quad (4.192)$$

so that the score is the distance to the nearest of our predicted quantiles. From this score, we may then form

$$\hat{q} = \text{Quantile} \left(\{s(x_i, y_i) | (x_i, y_i) \in \mathcal{D}_{cal}\}, \frac{\lceil (n+1)(1-\alpha) \rceil}{n} \right), \quad (4.193)$$

that is, \hat{q} is the $\frac{\lceil (n+1)(1-\alpha) \rceil}{n}$ -th quantile of the scores from our calibration set. Using these corrections, we then output the *conformalized* interval

$$C(x) = [\hat{t}_{\alpha/2}(x) - \hat{q}, \hat{t}_{1-\alpha/2}(x) + \hat{q}] \quad (4.194)$$

which we have guaranteed achieves the desired coverage on the holdout set. If \mathcal{D}_{cal} is sufficiently representative, then we will have achieved a rigorous interval for a coverage of α . The effect is that \hat{q} adjusts our previously estimated interval so that a positive \hat{q} widens the interval and a negative \hat{q} shrinks it in order to obtain the appropriate coverage.

How do we implement this in practice? Any model that we fit by minimizing the mean squared error (e.g. a Neural Network) can be agumented to instead fit the so-called *pinball*

loss

$$L_\gamma(\hat{t}_\gamma, y) = \begin{cases} (y - \hat{t}_\gamma)\gamma; & \hat{t}_\gamma < y \\ (\hat{t}_\gamma - y)(1 - \gamma); & \hat{t}_\gamma \geq y \end{cases} \quad (4.195)$$

For $\gamma = 0.5$, this reduces to the standard ℓ_1 loss $\ell_1(\hat{t}, y) = |\hat{t} - y|/2$ which encourages models to learn the median (i.e. the 0.5 quantile).

NOTE: Add plot of pinball loss for reference

4.7.2 Conformalizing Scalar Uncertainty Estimates

Another common method for producing uncertainty estimates is to simultaneously predict the mean and standard deviation of the target distribution by assuming a Gaussian shape. More generally, we may seek to estimate a model uncertainty directly by first training a model, $\hat{f}(x)$, to predict the target y , and then train a second model $\hat{r}(x)$ to predict the residual error, that is

$$\hat{r} = |y - \hat{f}(x)|. \quad (4.196)$$

In either case, one could then form a confidence interval as

$$\left[\hat{f}(x) - \hat{r}(x), \hat{f}(x) + \hat{r}(x) \right]. \quad (4.197)$$

Other reasonable approaches to extract a heuristic uncertainty from a trained model might include

- Compute the output variance of an ensemble model \hat{f}
- Compute the prediction variance obtained when randomly dropping out nodes from a neural network.
- Estimate the variance of \hat{f} resulting from small perturbations of the input x .

For all of these cases, we may call the uncertainty estimate $u(x)$ and form the score function

$$s(x, y) = \frac{|y - \hat{f}(x)|}{u(x)} \quad (4.198)$$

so that the score function s measures how well our uncertainty estimate $u(x)$ matches the actual error $|y - \hat{f}(x)|$. If the error is large our predicted uncertainty is small, then s will be large. Consequently, the score function tells us the correction we must apply to our estimate $u(x)$ in order to predict the actual model error, that is,

$$|y - \hat{f}(x)| = s(x, y)u(x). \quad (4.199)$$

If we now take \hat{q} to be the $\frac{\lceil (n+1)(1-\alpha) \rceil}{n}$ -th quantile as before, we are guaranteed that

$$p(s(x_{\text{cal}}, y_{\text{cal}}) \leq \hat{q}) \geq 1 - \alpha \quad (4.200)$$

An easy-to-use implementation of these methods is provided by the open-source Julia package `ConformalPrediction.jl` which is designed to integrate into the larger `MLJ.jl` machine learning framework. We will utilize these techniques throughout the remainder of this dissertation in order to establish uncertainty estimates for *all* of our regression models.

4.8 Scientific Machine Learning

4.8.1 Physics-Informed Neural Networks

4.8.2 Universal Differential Equations

4.9 Data Assimilation

NOTE: We should add further derivations from a Bayesian framework to fit in with what we've done for the Gaussian process regression and generative topographic maps. We should also add some more generic background to the topic of Data Assimilation, it's first uses in Meteorology,

The proper application of scientific models to make real-world predictions requires that we commit ourselves to a full accounting of all possible sources of uncertainty when reporting results. Further, the explosion of *big data* across scientific fields provides a plethora of observational data that our models are typically unequipped to incorporate when making predictions. The field of *Data Assimilation* addresses this problem by providing a family of techniques engineered to combine model output together with observational data whilst enabling a complete accounting of the sources of uncertainty. For chaotic systems in particular, data assimilation enables integration on long time scales that would be impossible via models alone. In this overview, we will follow the examples from (Ahmed et al., 2020).

Data assimilation can be cleanly developed in the framework of discrete dynamical systems. Since, at the end of the day, all of our scientific models must be discretized to be evaluated numerically, this is a reasonable course of action. Our goal is to find the best prediction for the system state vector u that combines our model predictions, also known as *forecasts*, with observational data. Model predictions are described via the discrete update equation:

$$u_{k+1} = \mathcal{M}(u_k; \theta) \quad (4.201)$$

For ODE systems, \mathcal{M} represents the time integration scheme for a model like

$$\frac{du}{dt} = f(u, t; \theta), \quad (4.202)$$

in other words, a particular choice of ODE integration scheme (Runge Kutta for example) used to evolve the state vector u from time u_k to u_{k+1} .

To measure the performance of our assimilation scheme, we denote the *true* value of the state vector as $u^{(t)}$. The output of our model is denoted $u^{(b)}$ (b superscript for *background*). The discrepancy between the true value and our forecast is denoted $\xi^{(b)} = u^{(t)} - u^{(b)}$ characterizing the extent to which our model prediction is imperfect. The observations of our system are denoted by $w_k = w(t_k)$. These observations do not necessarily need to be components

of the state vector u , but rather, are related to it via the *observation function*, $h : u_k \mapsto w_k$. For example, one may attempt to predict surface sea temperatures using data assimilation with data from satellite observations. The function h would then be the Stefan-Boltzmann relating the measured spectra to temperature. However, real world data is *also* imperfect, which we must take into account let we over constrain our model with poor quality data. We write

$$w_k = h(u_k) + \xi_k^{(m)} \quad (4.203)$$

where $\xi_k^{(m)}$ denotes this measurement error.

Given our model predictions $u_k^{(b)}$ and observations w_k , we seek to obtain the *optimal* or best-possible prediction called the **analysis**, $u^{(a)}$. Despite our care, this analysis will still be imperfect, so we further define the analysis error as

$$\xi^{(a)} = u^{(t)} - u^{(a)} \quad (4.204)$$

In summary, we have defined the following relevant quantities (at time t_k):

$$u_k^{(t)} \in \mathbb{R}^n \quad \text{the true state vector} \quad (4.205)$$

$$u_k^{(b)} \in \mathbb{R}^n \quad \text{the model forecast} \quad (4.206)$$

$$u_k^{(a)} \in \mathbb{R}^n \quad \text{the analysis} \quad (4.207)$$

$$w_k \in \mathbb{R}^m \quad \text{the observation vector} \quad (4.208)$$

$$\xi^{(b)} \in \mathbb{R}^n \quad \text{the model forecast error} \quad (4.209)$$

$$\xi^{(m)} \in \mathbb{R}^m \quad \text{the observation noise vector} \quad (4.210)$$

$$\xi^{(a)} \in \mathbb{R}^n \quad \text{the analysis error} \quad (4.211)$$

$$\xi^{(p)} \in \mathbb{R}^n \quad \text{the process noise if we used our model on the true state} \quad (4.212)$$

$$\mathcal{M} : \mathbb{R}^n \rightarrow \mathbb{R}^n \quad \text{the discrete model evolution function} \quad (4.213)$$

$$f : \mathbb{R}^n \rightarrow \mathbb{R}^n \quad \text{differential equation model (the right hand side)} \quad (4.214)$$

$$h : \mathbb{R}^n \rightarrow \mathbb{R}^m \quad \text{observation function} \quad (4.215)$$

To move forward, we now establish the following (prior) assumptions about the distribution of errors in each component in order to make it possible to derive a closed form for the final analysis. We require

$$\mathbb{E}[\xi_k^{(b)}] = 0 \qquad \mathbb{E}[\xi_k^{(b)}(\xi_j^{(b)})^T] = 0 \text{ for } k \neq j \qquad (4.216)$$

$$\mathbb{E}[\xi_k^{(m)}] = 0 \qquad \mathbb{E}[\xi_k^{(m)}(\xi_j^{(m)})^T] = 0 \text{ for } k \neq j \qquad (4.217)$$

$$\mathbb{E}[\xi_k^{(b)}(u_0)^T] = 0 \qquad \mathbb{E}[\xi_k^{(m)}(u_0)^T] = 0 \qquad (4.218)$$

$$\mathbb{E}[\xi_k^{(b)}\xi_j^{(m)}] = 0 \qquad (4.219)$$

or in words, the errors in our model and observation are unbiased (e.g. mean zero) and uncorrelated.

We also define the error covariance matrices

$$Q_k := \mathbb{E}[\xi_k^{(p)}(\xi_k^{(p)})^T] \qquad (4.220)$$

$$R_k := \mathbb{E}[\xi_k^{(m)}(\xi_k^{(m)})^T] \qquad (4.221)$$

$$B_k := \mathbb{E}[\xi_k^{(b)}(\xi_k^{(b)})^T] \qquad (4.222)$$

which we will use in our consideration of the final error of our analysis.

4.9.1 Kalman Filter

Given some model for the error covariance matrices Q_k and R_k , we would like a method that propagates *both* our model *and* the errors forward. This way we may guarantee that the accuracy of our analysis doesn't come at the cost of higher uncertainty.

The original implementation of the Kalman filter was for strictly linear systems, in particular as a filter for signal processing. We will first develop the analysis for this simplified case and then will generalize to the *Extended Kalman Filter* (EKF) that can handle fully nonlinear situations.

In the linear case, our system may be written as

$$u_{k+1}^{(t)} = M_k u_k^{(t)} + \xi_{k+1}^{(p)} \quad (4.223)$$

$$w_k = H_k u_k^{(t)} + \xi_k^{(m)} \quad (4.224)$$

where M_k and H_k are now matrices defining the linear problem.

The goal of the Kalman filter is to derive the analysis $u^{(a)}$ which minimizes the trace of the analysis error covariance matrix (i.e. sum of squared errors):

$$\text{Tr}(P_k) := \mathbb{E}[(u_k^{(t)} - u_k^{(a)})^T (u_k^{(t)} - u_k^{(a)})] \quad (4.225)$$

Finding the analysis consists of two steps: the forecast step and the assimilation step.

Beginning with the forecasting step, assume we have the analysis at time t_k denoted $u_k^{(a)}$.

Then the forecast for time t_{k+1} is

$$u_{k+1}^{(b)} = M_k u_k^{(a)}, \quad (4.226)$$

or in words, we obtain the next prediction by evolving the previous analysis forward. The background error is therefore

$$\xi_{k+1}^{(b)} = u_{k+1}^{(t)} - u_{k+1}^{(b)} \quad (4.227)$$

$$= M_k u_k^{(t)} + \xi_{k+1}^{(p)} - M_k u_k^{(a)} \quad (4.228)$$

$$= M_k (u_k^{(t)} - u_k^{(a)}) + \xi_{k+1}^{(p)} \quad (4.229)$$

$$= M_k \xi_k^{(a)} + \xi_{k+1}^{(p)} \quad (4.230)$$

We may now evaluate the covariance matrix of our background estimate as:

$$B_{k+1} = \mathbb{E}[\xi_{k+1}^{(b)} (\xi_{k+1}^{(b)})^T] \quad (4.231)$$

$$= \mathbb{E} \left[\left(M_k \xi_k^{(a)} + \xi_{k+1}^{(p)} \right) \left(M_k \xi_k^{(a)} + \xi_{k+1}^{(p)} \right)^T \right] \quad (4.232)$$

$$(4.233)$$

If we presume that $\mathbb{E}[\xi_k^{(b)}(\xi_{k+1}^{(p)})^T] = 0$, then the cross terms vanish and we are left with

$$\boxed{B_{k+1} = M_k P_k M_k^T + Q_{k+1}} \quad (4.234)$$

Thus we now have the background (i.e forecast) estimate of the state at t_{k+1} and its covariance matrix. Given a measurement w_{k+1} at the same time with covariance matrix R_{k+1} , we may now perform the assimilation step where we fuse the two sources of information to obtain $u_{k+1}^{(a)}$ and P_{k+1} .

Let's suppose that the analysis has the form

$$u_{k+1}^{(a)} = \nu + K_{k+1} w_{k+1} \quad (4.235)$$

for some vector $\nu \in \mathbb{R}^n$ and matrix $K_{k+1} \in \mathbb{R}^{m \times n}$. In a perfect world, we would have $\mathbb{E}[u_k^{(t)} - u_k^{(a)}] = 0$. Therefore,

$$0 = \mathbb{E}[u_k^{(t)} - u_k^{(a)}] \quad (4.236)$$

$$= \mathbb{E}[(u_k^{(b)} + \xi_k^{(b)}) - (\nu + K_k w_k)] \quad (4.237)$$

$$= \mathbb{E}[(u_k^{(b)} + \xi_k^{(b)}) - (\nu + K_k H_k u_k^{(t)} + K_k \xi_k^{(m)})] \quad (4.238)$$

$$= \mathbb{E}[u_k^{(b)}] + \mathbb{E}[\xi_k^{(b)}] - \mathbb{E}[\nu] - K_k H_k \mathbb{E}[u_k^{(t)}] - K_k \mathbb{E}[\xi_k^{(m)}] \quad (4.239)$$

$$= u_k^{(b)} + 0 - \nu - K_k H_k u_k^{(b)} - 0 \quad (4.240)$$

$$= u_k^{(b)} - \nu - K_k H_k u_k^{(b)} \quad (4.241)$$

$$\Rightarrow \nu = u_k^{(b)} - K_k H_k u_k^{(b)} \quad (4.242)$$

which we now substitute to obtain

$$\boxed{u_k^{(a)} = u_k^{(b)} + K_k (w_k - H_k u_k^{(b)})} \quad (4.243)$$

Now that we know the form for the analysis we may derive the optimal matrix K_k by optimization of P_k . We have

$$\xi_k^{(a)} = u_k^{(t)} - u_k^{(a)} \quad (4.244)$$

$$= M_{k-1}u_{k-1}^{(t)} + \xi_k^{(p)} - u_k^{(b)} - K_k \left(w_k - H_k u_k^{(b)} \right) \quad (4.245)$$

$$= M_{k-1}u_{k-1}^{(t)} + \xi_k^{(p)} - M_{k-1}u_{k-1}^{(a)} - K_k \left(H_k u_k^{(t)} + \xi_k^{(m)} - H_k u_k^{(b)} \right) \quad (4.246)$$

$$= M_{k-1}u_{k-1}^{(t)} + \xi_k^{(p)} - M_{k-1}u_{k-1}^{(a)} - K_k H_k u_k^{(t)} - K_k \xi_k^{(m)} + K_k H_k u_k^{(b)} \quad (4.247)$$

$$= M_{k-1}u_{k-1}^{(t)} + \xi_k^{(p)} - M_{k-1}u_{k-1}^{(a)} - K_k H_k u_k^{(t)} - K_k \xi_k^{(m)} + K_k H_k u_k^{(b)} \quad (4.248)$$

$$= \left\{ M_{k-1}(\xi_{k-1}^{(a)} + u_{k-1}^{(a)}) + \xi_k^{(p)} - M_{k-1}u_{k-1}^{(a)} \right\} - K_k H_k u_k^{(t)} - K_k \xi_k^{(m)} + K_k H_k u_k^{(b)} \quad (4.249)$$

$$= \left\{ M_{k-1}\xi_{k-1}^{(a)} + \xi_k^{(p)} \right\} - K_k H_k u_k^{(t)} + K_k H_k u_k^{(b)} - K_k \xi_k^{(m)} \quad (4.250)$$

$$= M_{k-1}\xi_{k-1}^{(a)} + \xi_k^{(p)} - K_k H_k (M_{k-1}u_{k-1}^{(t)} + \xi_k^{(b)}) + K_k H_k u_k^{(b)} - K_k \xi_k^{(m)} \quad (4.251)$$

$$= M_{k-1}\xi_{k-1}^{(a)} + \xi_k^{(p)} - K_k H_k M_{k-1}(\xi_{k-1}^{(a)} + u_{k-1}^{(a)}) - K_k H_k \xi_k^{(b)} + K_k H_k u_k^{(b)} - K_k \xi_k^{(m)} \quad (4.252)$$

$$= M_{k-1}\xi_{k-1}^{(a)} + \xi_k^{(p)} - K_k H_k M_{k-1}(\xi_{k-1}^{(a)} + u_{k-1}^{(a)}) - K_k H_k \xi_k^{(b)} + K_k H_k M_{k-1}u_{k-1}^{(a)} - K_k \xi_k^{(m)} \quad (4.253)$$

$$= M_{k-1}\xi_{k-1}^{(a)} + \xi_k^{(p)} - K_k H_k M_{k-1}\xi_{k-1}^{(a)} - K_k H_k \xi_k^{(b)} - K_k \xi_k^{(m)} \quad (4.254)$$

$$= (I - K_k H_k)(M_{k-1}\xi_{k-1}^{(a)} - \xi_k^{(p)}) - K_k \xi_k^{(m)} \quad (4.255)$$

$$(4.256)$$

and therefore the covariance matrix is

$$P_k = \mathbb{E}[\xi_k^{(a)}(\xi_k^{(a)})^T] \quad (4.257)$$

$$= \mathbb{E}\left[\left((I - K_k H_k)(M_{k-1}\xi_{k-1}^{(a)} - \xi_k^p) - K_k \xi_k^{(m)}\right)\left((I - K_k H_k)(M_{k-1}\xi_{k-1}^{(a)} - \xi_k^p) - K_k \xi_k^{(m)}\right)^T\right] \quad (4.258)$$

$$= (I - K_k H_k)M_{k-1}\mathbb{E}[\xi_{k-1}^{(a)}(\xi_{k-1}^{(a)})^T]M_{k-1}^T(I - K_k H_k)^T \quad (4.259)$$

$$\begin{aligned} &+ (I - K_k H_k)\mathbb{E}[\xi_k^{(p)}(\xi_k^{(p)})^T](I - K_k H_k)^T \\ &- K_k\mathbb{E}[\xi_k^{(m)}(\xi_k^{(m)})^T]K_k^T \\ &= (I - K_k H_k)B_k(I - K_k H_k)^T - K_k R_k K_k^T \end{aligned} \quad (4.260)$$

Now all that remains is to derive the specific form for K_k . The Kalman filter is defined as that K_k which minimizes the sum of squared analysis errors, i.e. the trace of the analysis error covariance matrix. Therefore, the following identities from matrix calculus will be useful:

$$\nabla_A \text{tr}(AB) = B^T \quad (4.261)$$

$$\nabla_A \text{tr}(BA^T) = B \quad (4.262)$$

$$\nabla_A \text{tr}(ABA^T) = AB^T + AB \quad (4.263)$$

$$(4.264)$$

from which we obtain

$$0 = \nabla_{K_k} \text{tr}(P_k) \quad (4.265)$$

$$= \nabla_{K_k} \left\{ B_k - B_k H_k^T K_k^T - K_k H_k B_k + K_k H_k B_k H_k^T B_k^T - K_k R_k K_k^T \right\} \quad (4.266)$$

$$= -B_k H_k^T - (H_k B_k)^T + K_k [H_k B_k H_k^T + (H_k B_k H_k^T)^T - R_k + R_k^T] \quad (4.267)$$

$$= -2B_k H_k^T + 2K_k (H_k B_k H_k^T - R_k) \quad (4.268)$$

$$\Rightarrow K_k = B_k H_k^T [H_k B_k H_k^T - R_k]^{-1} \quad (4.269)$$

we now substitute this result to obtain a simplified form for P_k :

$$P_k = (I - K_k H_k) B_k (I - K_k H_k)^T + K_k R_k K_k^T \quad (4.270)$$

$$= (I - K_k H_k) B_k - (I - K_k H_k) B_k (K_k H_k)^T + K_k R_k K_k^T \quad (4.271)$$

$$= (I - K_k H_k) B_k - \left\{ (I - K_k H_k) B_k (K_k H_k)^T + K_k R_k K_k^T \right\} \quad (4.272)$$

$$= (I - K_k H_k) B_k - \left\{ (I - K_k H_k) B_k H_k^T K_k^T + K_k R_k K_k^T \right\} \quad (4.273)$$

$$= (I - K_k H_k) B_k - \left\{ (I - K_k H_k) B_k H_k^T + K_k R_k \right\} K_k^T \quad (4.274)$$

$$= (I - K_k H_k) B_k - \left\{ B_k H_k^T - K_k (H_k B_k H_k^T + R_k) \right\} K_k^T \quad (4.275)$$

$$= (I - K_k H_k) B_k - \left\{ B_k H_k^T - B_k H_k^T \right\} K_k^T \quad (4.276)$$

$$= (I - K_k H_k) B_k \quad (4.277)$$

where we have used the fact that covariance matrices are symmetric.

Now that we have all of the pieces, let's summarize the procedure:

1. **Initialization:** We must set the system to some initial condition. This means we must define u_0^a and P_0 . We must also come up with a model for the process noise covariance Q_k and measurement error covariance R_k .
2. **Forecast Step:** Starting with the analysis state vector $u_{k-1}^{(a)}$ and covariance P_{k-1} , we apply the model time evolution operator M_{k-1} to obtain the predicted state $u_k^{(b)}$ and covariance B_k at the next time step.

$$u_k^{(b)} = M_{k-1} u_{k-1}^{(a)} \quad (4.278)$$

$$B_k = M_{k-1} P_{k-1} M_{k-1}^T + Q_k \quad (4.279)$$

3. **Assimilation Step:** Combine our observations w_k and their associated uncertainties contained in R_k together with the background prediction $u_k^{(b)}$ and its error covariance

matrix B_k to obtain the analysis state $u_k^{(a)}$ and error covariance P_k .

$$K_k = B_k H_k^T \left[H_k B_k H_k^T - R_k \right]^{-1} \quad (4.280)$$

$$u_k^{(a)} = u_k^{(b)} + K_k (w_k - H_k u_k^{(b)}) \quad (4.281)$$

$$P_k = (I - K_k H_k) B_k \quad (4.282)$$

4.9.2 Extended Kalman Filter

Given the nonlinear nature of many scientific models, it is desirable to extend out notion of the *Kalman Filter* to be able to handle nonlinear models $f(\cdot)$ (and by extension, their update function $\mathcal{M}(\cdot)$), and nonlinear observation functions $h(\cdot)$. This can be accomplished so long as these functions are sufficiently smooth (C^1 to be precise) so as to admit valid Taylor approximations to first order. That is,

$$\mathcal{M}(u_k) \approx \mathcal{M}(u_k^{(a)}) + D_M(u_k^{(a)}) \xi_k^{(a)} \quad h(u_k) \approx h(u_k^{(b)}) + D_h(u_k^{(b)}) \xi_k^{(b)} \quad (4.283)$$

$$D_M := \left[\frac{\partial \mathcal{M}_i}{\partial u_j} \right] \quad D_h := \left[\frac{\partial h_i}{\partial u_j} \right] \quad (4.284)$$

where \mathcal{M}_i and h_i denote the i th component functions of \mathcal{M} and h respectively. For the so-called *Extended Kalman Filter* (EKF), approach. We assume that the analysis errors can be reasonably approximated to evolve linearly such that we can use the above *linearized* approximations to our model update and observation functions to reasonably model our uncertainties.

We therefore replace the (previously) linear functions M_k and H_k with their nonlinear Jacobian derived counterparts to obtain the following procedure

1. **Initialization:** To begin we must choose values for $u_0^{(a)}$ and P_0 . We must also provide models for Q_k and R_k .
2. **Forecast Step:**

$$u_k^{(b)} = \mathcal{M}(u_{k-1}^{(a)}) \quad (4.285)$$

$$B_k = D_M(u_{k-1}^{(a)}) P_{k-1} D_M^T(u_{k-1}^{(a)}) + Q_k \quad (4.286)$$

3. Assimilation Step:

$$K_k = B_k D_h^T(u_k^{(b)}) \left[D_h(u_k^{(b)}) B_k D_h^T(u_k^{(b)}) + R_k \right]^{-1} \quad (4.287)$$

$$u_k^{(a)} = u_k^{(b)} + K_k (w_k - h(u_k^{(b)})) \quad (4.288)$$

$$P_k = \left(I - K_k D_h(u_k^{(b)}) \right) B_k \quad (4.289)$$

Note that the most challenging piece in this implementation is to stably compute the model update Jacobian D_M , as this is not just the Jacobian of our continuous model $f(\cdot)$. For example, if we are modeling the state of the system via a set of ODEs, then D_M is the Jacobian of the output of a single integrator step with respect to the initial starting state. This can pose significant computational challenges where adaptive solvers are desired as we must be able to back-propagate out derivative information for all evaluations of the function $f(\cdot)$ taken during the step.

4.9.3 continuous-discrete Extended Kalman Filter

So far in our discussion of data assimilation techniques we have limited ourselves to discretized realizations of continuous models and measurements. This allowed us to derive reasonable update equations for the error covariance matrix and analysis by considering the changes that occur by application of our model evolution operator \mathcal{M} . In practice though the case tends to be slightly different: the underlying dynamics are understood to evolve continuously despite our access to finite time measurements at some frequency. For this reason, it would be preferable to utilize our knowledge of the model covariance Q together with the continuous dynamics of the model itself to allow the background covariance matrix to evolve continuously between measurements. Then, only when we have access to new data to we need to perform a discrete update by computing the Kalman gain and generating a new analysis as before. As we shall see, this strategy yields an augmented dynamical system

for which

$$\begin{cases} \frac{du}{dt} = f(u, t; \theta) + \xi^{(b)}(t) \\ \frac{P}{dt} = J_u(t)P(t) + P(t)J_u^T(t) + Q(t) \end{cases} \quad (4.290)$$

where $J_u = \frac{\partial f}{\partial u}$ is the Jacobian of the system with respect to the state vector u . To derive this new form for the Extended Kalman filter, let us first begin with a linear system for which *both* the underlying dynamics continuously, that is

$$\begin{cases} \dot{u} = Mu + \xi^{(b)} \\ w_k = Hu_k + \xi_k^{(m)} \end{cases} \quad (4.291)$$

where $\xi^{(b)}$ now represents a Weiner process (e.g. Gaussian white noise process) of covariance $Q(t)$. Presuming the model covariance is still uncorrelated with the measurement means that we now have:

$$\mathbb{E}[\xi^{(b)}(t)(\xi^{(b)}(\tau))^T] = Q(t)\delta(t - \tau) \quad (4.292)$$

$$\mathbb{E}[\xi^{(m)}(\xi^{(b)}(t))^T] = 0 \quad (4.293)$$

To move forward, we now need to establish the relationship between our previous Q_k and Q so that in the limit as $\Delta t \rightarrow 0$, we may obtain the new dynamical equations. If we understand Q_k to be the model error *accumulated* over the integration window, then it follows that

$$\begin{aligned} Q_k &\approx \int \int d\zeta d\eta \mathbb{E}[\xi^{(b)}(\zeta)(\xi^{(b)}(\eta))^T] \\ &= \int \int d\zeta d\eta Q(\zeta)\delta(\zeta - \eta) \\ &= \int_{t_{k-1}}^{t_k} Q(\zeta) d\zeta \\ &\approx Q(t_k)\Delta t \end{aligned} \quad (4.294)$$

This tells us all that we need! For small enough Δt , we may approximate our continuous dynamics by

$$\frac{u_{k+1} - u_k}{\Delta t} = Mu_k + \xi_k^{(b)} \quad (4.295)$$

and therefore

$$u_{k+1} = (I - \Delta t M) u_k + \xi_k^{(b)} \quad (4.296)$$

from which we can identify our linear M_k matrix from the original discrete Kalman filter as $(I + \Delta t M)$ in the discretized version of this new continuous model. Substituting this in to our expression for the forecast step yields

$$B_{k+1} = (I + \Delta t M) P_k (I + \Delta t M)^T + Q \Delta t. \quad (4.297)$$

Expanding this expression and keeping only terms of $\mathcal{O}(\Delta t)$, we find

$$\begin{aligned} B_{k+1} &= P_k + \Delta t M P_k + P_k M^T \Delta t + M P_k M^T \Delta t^2 + Q \Delta t \\ &\approx P_k + \Delta t M P_k + \Delta t P_k M^T + Q \Delta t \end{aligned} \quad (4.298)$$

As we shrink Δt to zero between observations, there is no longer a distinction between the background covariance B_k and the analysis covariance P_k so that we may write rearrange and take the limit

$$\lim_{\Delta t \rightarrow 0} \frac{P_{k+1} - P_k}{\Delta t} = \lim_{\Delta t \rightarrow 0} M P_k + P_k M^T + Q \quad (4.299)$$

which evaluates to

$$\boxed{\frac{d}{dt} P = M P(t) + P(t) M^T + Q(t)} \quad (4.300)$$

To *extend* this linear version to the case of nonlinear model and observation functions, we linearize our nonlinear model so that $M \mapsto J_u = \partial f / \partial u$. This then leads to the continuous-discrete Extended Kalman Filter for which the procedure is as follows

1. **Initialization:** To begin we choose initial values for $u_0^{(a)}$ and P_0 . We must also provide initial covariance values R_0 and $Q(0)$.
2. **Forecast Step:** Integrate the dynamical system

$$\begin{cases} \frac{du}{dt} = f(u, t; \theta) \\ \frac{dP}{dt} = J_u P(t) + P(t) J_u^T + Q(t) \end{cases} \quad (4.301)$$

from t_{k-1} to t_k to obtain u_k and P_k

3. **Assimilation Step:** Compute the Kalman gain and update the state vector and associated covariance matrix according to

$$K_k = P_k D_h^T(u_k) [D_h(u_k) P_k D_h^T(u_k) + R_k]^{-1} \quad (4.302)$$

$$u_k \mapsto u_k + K_k(w_k - h(u_k)) \quad (4.303)$$

$$P_k \mapsto (I - K_k D_h(u_k)) P_k \quad (4.304)$$

4.9.4 3d-Var

For the *Kalman Filter* and the *EKF*, we derived an optimal way to combine observation with simulation so as to minimize the trace of the analysis error covariance matrix, P_k . An alternative approach is to recast the problem as a pure optimization problem where rather than finding a filter K_k that will add an innovation to $u_k^{(b)}$ to obtain the analysis $u_k^{(a)}$, we obtain the analysis by directly optimizing a cost function

$$J(u) = \frac{1}{2} (w - h(u))^T R^{-1} (w - h(u)) + \frac{1}{2} (u - u^{(b)})^T B^{-1} \frac{1}{2} (u - u^{(b)}) \quad (4.305)$$

which we can justify as coming from the joint probability distribution (assuming Gaussian errors)

$$\mathcal{P}(u|w) = C \exp \left(-\frac{1}{2} (u - u^{(b)})^T B^{-1} \frac{1}{2} (u - u^{(b)}) \right) \cdot \exp \left(-\frac{1}{2} (w - h(u))^T R^{-1} (w - h(u)) \right) \quad (4.306)$$

with model error covariance B and measurement error covariance R as before. This is clearly a *very strong assumption*.

To optimize $J(u)$, we begin by taking it's gradient.

$$\nabla_u J(u) = -D_h^T R^{-1} (w - h(u)) + B^{-1} (u - u^{(b)}) \quad (4.307)$$

Thus, finding the analysis $u^{(a)}$ amounts to solving the system

$$a - D_h^T R^{-1} (w - h(u^{(a)})) + B^{-1} (u^{(a)} - u^{(b)}) = 0 \quad (4.308)$$

As for Kalman filtering, let's begin with the assumption that our model and observation function are linear. Suppose that we have $h(u) = Hu$ so that $D_h(u) = H$. It follows that

$$D_h^T R^{-1}(w - Hu^{(a)}) = B^{-1}(u^{(a)} - u^{(b)}) \quad (4.309)$$

$$D_h^T R^{-1}w - D_h^T R^{-1}Hu^{(a)} = B^{-1}u^{(a)} - B^{-1}u^{(b)} \quad (4.310)$$

$$(D_h^T R^{-1}H + B^{-1})u^{(a)} = D_h^T R^{-1} + B^{-1}u^{(b)} \quad (4.311)$$

$$(H^T R^{-1}H + B^{-1})u^{(a)} = H^T R^{-1} + B^{-1}u^{(b)} \quad (4.312)$$

Thus we see that the analysis is given by

$$u^{(a)} = u^{(b)} + BH^T (R + HB^T H)^{-1} (w - Hu^{(b)}) \quad (4.313)$$

which agrees with what we found for the Linear Kalman Filter.

Moving on to the non-linear case, we can expand h about an initial guess $u^{(c)}$ which we will later choose to be $u^{(b)}$ for convenience.

$$h(u^{(a)}) \approx h(u^{(c)}) + D_h(u^{(c)})\Delta u \quad (4.314)$$

Using this, we have

$$D_h^T(u^{(a)})R^{-1}(w - h(u^{(a)})) = B^{-1}(u^{(a)} - u^{(b)}) \quad (4.315)$$

$$D_h^T(u^{(a)})R^{-1}(w - h(u^{(c)}) - D_h(u^{(c)})\Delta u) \approx B^{-1}(u^{(c)} + \Delta u - u^{(b)}) \quad (4.316)$$

$$D_h^T(u^{(c)})R^{-1}(w - h(u^{(c)}) - D_h(u^{(c)})\Delta u) \approx B^{-1}(u^{(c)} + \Delta u - u^{(b)}) \quad (4.317)$$

which we now solve for the update Δu to obtain the linear system

$$(B^{-1} + D_h^T(u^{(c)})R^{-1}D_h(u^{(c)}))\Delta u = B^{-1}(u^{(b)} - u^{(c)}) + D_h^T(u^{(c)})R^{-1}(w - h(u^{(c)})) \quad (4.318)$$

Thus we have the following prescription

1. To begin, take $u^{(c)} = u^{(b)}$.

2. Solve the system

$$(B^{-1} + D_h^T(u^{(c)})R^{-1}D_h(u^{(c)})) \Delta u = B^{-1}(u^{(b)} - u^{(c)}) + D_h^T(u^{(c)})R^{-1}(w - h(u^{(c)})) \quad (4.319)$$

to obtain Δu

3. Update your guess using your favorite optimization algorithm. For example, in steepest descent, choose a learning rate η and set

$$u_{\text{new}}^{(c)} = u_{\text{prev}}^{(c)} - \eta \Delta u \quad (4.320)$$

4. Repeat the procedure until $|u_{\text{new}}^{(c)} - u_{\text{prev}}^{(c)}|$ converges to a desired tolerance.

In both the linear and nonlinear case, it should be noted that we have not added time indices to our state vectors. This is an indication that the 3d-var procedure is performed *at every time where you have observation data*.

4.9.5 4d-Var

The *3d-Var* algorithm attempts to optimize a cost function point-by-point to obtain the ideal analysis at each time where we have observation data. This can become computationally expensive as we require model evaluations *and* an optimization routine for every observation point. An alternative approach is to simultaneously optimize across all observations in order to obtain the ideal *initial condition* which achieves the best model fit. In other words, this amounts to parameter fitting for our model where we think of the initial conditions as the parameters. The difference is two-fold: We extend the usual ℓ_2 function into a quadratic form utilizing the observation error covariance matrix R_k . We also typically only have a small number of observables related to the state vector (potentially non-linearly) and therefore

must transform the state vector u_k into the observation space via h before computing the loss against our observations w_k . This results in a loss function of the form

$$\begin{aligned} J(u_0) &= \frac{1}{2} \left(u_0 - u_0^{(b)} \right)^T B^{-1} \left(u_0 - u_0^{(b)} \right) + \frac{1}{2} \sum_k (w_k - h(u_k))^T R_k^{-1} (w_k - h(u_k)) \quad (4.321) \\ &= J_b(u_0) + J_m(u_0). \end{aligned}$$

Note that the first term is usefull if we already have an initial guess $u_0^{(b)}$ for the inital condition in mind. If we do not have one, we may ommit this term.

As before, we now want to optimize this cost function. To do so, we first observe that

$$u_k = \mathcal{M}^{(k)}(u_0; \theta) \quad (4.322)$$

It is easy to obtain the gradient of J_b so we shall focus on the second term. We find that

$$\nabla_{u_0} J_m = \nabla_{u_0} \left\{ \sum_k \frac{1}{2} (w_k - h(u_k))^T R_k^{-1} (w_k - h(u_k)) \right\} \quad (4.323)$$

$$= - \sum_k \left[\frac{\partial}{\partial u_0} h(\mathcal{M}^{(k-1)}(u_0)) \right]^T R_k^{-1} (w_k - h(u_k)) \quad (4.324)$$

$$= - \sum_k [D_h(u_k) D_M(u_{k-1}) D_M(u_{k-2}) \cdots D_M(u_0)]^T R_k^{-1} (w_k - h(u_k)) \quad (4.325)$$

$$= - \sum_k [D_M^T(u_0) D_M^T(u_1) \cdots D_M^T(u_{k-1}) D_h^T(u_k)] R_k^{-1} (w_k - h(u_k)) \quad (4.326)$$

With this in hand, the procedure is nearly identical to 3d-var:

1. Integrate your model forward to obtain $\{u_k\}$
2. Evaluate each of the $D_M^T(u_{k-1:0})$ and $D_h(u_k)$.
3. Using these values, compute $\nabla J_m(u)$
4. Set $u_0^{(new)} = u_0^{(prev)} - \eta \nabla J(u_0^{(prev)})$
5. Stop when $|u_0^{(new)} - u_0^{(prev)}|$ converges to your desired tolerance.

You can of course substitute another optimization scheme after step 3.

There are a few further observations that we can make about this procedure. The first, is that should we feel our model $f(\cdot)$ is not sufficient to fit the entire time series without the periodic corrections of the Kalman filter (perhaps there is some missing physics we haven't captured in our model), then it may be ideal to restrict the optimization to a subset of the full time series of observations. The second observation is that the above formulation has assumed the adjoint D_M^T can be computed easily. We can take advantage of the methods from the *Adjoint Sensativity Analysis* of ODEs to take advantage of modern methods for automatic differentiation of ODE solutions. Finally, it should be noted that the $4d - var$ algorithm yields the optimal starting conditions assuming our model is perfect. At the sacrifice of smoothness, it is often desirable to first obtain the ideal initial condition with $4d - var$ and *then* employ a filtering method like the EKF to obtain the final analysis.

CHAPTER 5
AN AUTONOMOUS ROBOTIC TEAM FOR THE RAPID
CHARACTERIZATION OF NOVEL ENVIRONMENTS

- 5.1 Rapid Georectification and Processing of Pushbroom Hyperspectral Imagery**
- 5.2 Supervised Regression with Uncertainty Quantification**
- 5.3 Methods for Unsupervised Classification of Hyperspectral Scenes in Novel Environments**

CHAPTER 6

A CHEMICAL DATA ASSIMILATION FRAMEWORK FOR INDOOR AIR QUALITY

6.1 Characterization of Photolysis

6.2 HEART Chamber Sensing System

6.3 Chemical Data Assimilation

6.4 An Evaluation of Photocatalytic Ionization

- sensor fusion
- photolysis
- docker ingestion framework
- Master Chemical Mechanism
- CRI Mechanism
- AutoChem
- data assimilation
- Overview of all sensor in sensor matrix
- Overview of measurement capabilities (list of species, uncertainty levels, etc...)
- Overview of containerized data acquisition pipeline
- NodeRed
- InfluxDB

- Grafana
- Quarto
- Automatic Alerts
- Automatic Reports
- MCM Implementation in Julia
- Direct computation of Photolysis rates
- Combination with Dr. Lary's AutoChem
- Addition of Ion Chemistry from MIT Lightning disertation
- Visualization of chemical cycles
- SciML methods to infer below detection limits
- Ion Chemistry
- Indoor Air Quality
- Photocatalytic Ionization

CHAPTER 7

A DISTRIBUTED NETWORK OF LOW COST AIR QUALITY SENSORS

7.1 MINTS Air Quality Network

7.1.1 Central Nodes

7.1.2 LoRa Nodes

7.1.3 MQTT

7.2 Real Time Dashboards via Containerization

7.3 Making Data Publicly Accessible via S3 and the Open Storage Network

discuss Rclone and using OSN to make data highly available

- LoRa wan devices
- Docker, NodeRed, InfluxDB, Grafana
- Hamiltonian NN stuff
- neural ode
- TDA

Sensor Network + SciML Evaluation of local chaos in SharedAirDFWNetwork

- This gives me an excuse to work with the sensor data
- Train GTM, SOM, and Variational Autoencoder to produce lower dimensional representation of all data from a central node, e.g. in \mathbb{R}^2 .
- For VAE, test a range of dimensions from the number of sensors down to 2 (better for visualization)

- Analyze the variety of methods from [DataDrivenDiffEq.jl](https://docs.sciml.ai/DataDrivenDiffEq.jl) to infer dynamics in the low dimensional space
- Can we infer some kind of Hamiltonian from the data and do a HamiltonianNN approach?
- Start of with a standard kinetic-energy style Hamiltonian e.g. $\sum_i \frac{1}{2} \dot{x}_i^2$ where x_i is the
- use DataDrivenDiffEq approach to learn the associated potential energy term
- alternatively, attempt to capture diurnal cycle (or other relevant time scales) by *learning* coordinate representation that forces dynamics to be uncoupled harmonic oscillators a la Hamilton-Jacobi theory.
- Test if this hamiltonian NN model can then be transfered to another central node with an appropriate shift in the "total energy"
- Attempt to analyze the 2d data to infer Koopman operator. We should treat the original sensor values as observables on which the learned koopman operator acts. This should be doable if the NN is just a function.
- use DMD approach to identify a "forcing" coordinate that can identify when we switch nodes as in (Brunton et al., 2017)
- video on Physics Informed DMD
- Deep Learning to Discover Coordinates for Dynamics: Autoencoders and Physics Informed Machine Learning
- **NOTE:** we may need to impute missing values. We should do so with either my GPR code or with other ML methods + ConformalPrediction. Provided uncertainty estimates, we should then think about how to propagate errors through our analysis via Measurements.jl, IntervalArithmetic.jl.

CHAPTER 8

TIME SERIES METHODS FOR AIR QUALITY

8.1 Time-Series Methods for Uncertainty Quantification

8.1.1 Metrics for Representative Uncertainty of combined Pseudo-Observations

8.1.2 Uncertainty Quantification with Temporal Variograms

8.2 Physics Informed modeling techniques for Air Quality Data

8.2.1 Embedding Theorems

Discuss Taken's embedding theorem and other relevant information for the time-series work.

8.2.2 Koopman Theory

Give an overview of continuous and discrete Koopman operator theory.

8.2.3 HAVOK + SciML = Nonlinear Modeling with External Forcing

8.2.4 Hamiltonian Neural Networks

- Uncertainty Estimation Via Time Series Sampling
- Types Of Uncertainty
- Instrument Uncertainty
- Representativeness Uncertainty
- Variograms
- Mutual Information
- Auto-correlation

- Time Series Chaos
- What is Chaos?
- Lyapunov Exponents
- Fractal Dimension
- Koopman Operator Theory
- Time Series Modeling Methods
- Token-Hankel Delay Embeddings
- Embedding Theorems (are magic)
- Determination of Optimal Lag
- Determination of Intrinsic Dimension (kind of unnecessary given *large enough* embedding)
- DMD and HAVOK
- Hamiltonian Neural Networks
- Time Series Classification
- Considerations for Batching of Time Series for ML Models
- K-means Clustering
- Self Organizing Maps
- Generative Topographic Maps
- Chaos Classification via HAVOK

- Symplectic and Normal Gradients for HNN
- Motivating Example: Lorenz63 System
- Origin of Lorenz System
- Time Scale Analysis and Variography
- Embedding
- Modeling
- Classification
- Real Example 1: PM Data
- Origin of Lorenz System
- Time Scale Analysis and Variography
- Embedding
- Modeling
- Classification
- Real Example 2: Biometric Data
- Origin of Lorenz System
- Time Scale Analysis and Variography
- Embedding
- Modeling
- Classification

- Real Example 3: Stock Market Analysis
- Origin of Lorenz System
- Time Scale Analysis and Variography
- Embedding
- Modeling
- Classification

CHAPTER 9

LIMITATIONS AND FUTURE WORK

9.1 Robot Team

9.1.1 Super Resolution

Propose methods for spatial/temporal/spectral super-resolution. In particular, comment on upcoming satellite deployments like EnMAP and others which will provide hyperspectral imagery.

CHAPTER 10
CONCLUSIONS

UPDATE REQUIRED!!!

APPENDIX A
CHEMICAL REACTION MECHANISMS

UPDATE REQUIRED!

A.1 Simple Ion Mechanism

Here we can include the automatically generated documentation for the Ion Mechanism and any others.

A.1.1 Bimolecular Reactions

#	Bimolecular Reaction	Reaction Rate Coeff
1	$\text{Cl} + \text{HNO}_3 \longrightarrow \text{HCl} + \text{NO}_3$	$k = (2.0\text{e-}16)$
2	$\text{Br} + \text{H}_2 \longrightarrow \text{H} + \text{HBr}$	$k = (8.0\text{e-}11) \exp(-8970.0/T)(T/298.0)^{0.43}$
3	$\text{Br} + \text{O}_3 \longrightarrow \text{O}_2 + \text{BrO}$	$k = (1.7\text{e-}11) \exp(-800.0/T)$
4	$\text{Br} + \text{OH} \longrightarrow \text{O}({}^3\text{P}) + \text{HBr}$	$k = (4.56\text{e-}12) \exp(-8715.0/T)$

5	$\text{Br} + \text{CH}_4 \longrightarrow \text{HBr} + \text{CH}_3$	$k = (7.8\text{e-}11) \exp(-76330.0/T)$
6	$\text{Br} + \text{H}_2\text{O} \longrightarrow \text{OH} + \text{HBr}$	$k = 0$
7	$\text{Br} + \text{HCO} \longrightarrow \text{CO} + \text{HBr}$	$k = (2.8\text{e-}10) \exp(-800.0/T)$
8	$\text{Br} + \text{HO}_2 \longrightarrow \text{O}_2 + \text{HBr}$	$k = (1.5\text{e-}11) \exp(-600.0/T)$
9	$\text{Br} + \text{N}_2\text{O} \longrightarrow \text{N}_2 + \text{BrO}$	$k = (3.32\text{e-}10) \exp(-154800.0/T)$
10	$\text{Br} + \text{NO}_3 \longrightarrow \text{BrO} + \text{NO}_2$	$k = (1.6\text{e-}11)$
11	$\text{Br} + \text{H}_2\text{O}_2 \longrightarrow \text{HBr} + \text{HO}_2$	$k = (1.0\text{e-}11) \exp(-3000.0/T)$

12	$\text{Br} + \text{HCHO} \longrightarrow \text{HBr} + \text{HCO}$	$k = (1.7\text{e-}11) \exp(-800.0/T)$
13	$\text{Br} + \text{CH}_3\text{OH} \longrightarrow$ $\text{HBr} + \text{HO}_2 + \text{HCHO}$	$k = (7.59\text{e-}13) \exp(-3119.0/T)$
14	$\text{Br} + \text{OClO} \longrightarrow \text{BrO} + \text{ClO}$	$k = (2.6\text{e-}11) \exp(-1300.0/T)$
15	$\text{Br} + \text{Cl}_2\text{O}_2 \longrightarrow \text{BrCl} + \text{ClOO}$	$k = (3.0\text{e-}12)$
16	$\text{Br} + \text{CH}_3\text{OOH} \longrightarrow \text{HBr} + \text{CH}_3\text{O}_2$	$k = (2.63\text{e-}12) \exp(-1610.0/T)$
17	$\text{Br} + \text{BrONO}_2 \longrightarrow \text{Br}_2 + \text{NO}_3$	$k = (6.3\text{e-}11) \exp(213.0/T)$
18	$\text{BrO} + \text{BrO} \longrightarrow$ $\text{Br} + \text{O}_2 + \text{Br}$	$k = (9.23\text{e-}13) \exp(250.0/T)$

19	$\text{BrO} + \text{BrO} \longrightarrow \text{O}_2 + \text{Br}_2$	$k = (1.8\text{e-}13) \exp(250.0/T)$
20	$\text{BrO} + \text{ClO} \longrightarrow \text{Br} + \text{OClO}$	$k = (1.6\text{e-}12) \exp(430.0/T)$
21	$\text{BrO} + \text{ClO} \longrightarrow \text{Br} + \text{ClOO}$	$k = (2.9\text{e-}12) \exp(220.0/T)$
22	$\text{BrO} + \text{ClO} \longrightarrow \text{O}_2 + \text{BrCl}$	$k = (5.8\text{e-}13) \exp(170.0/T)$
23	$\text{BrO} + \text{HO}_2 \longrightarrow \text{O}_3 + \text{HBr}$	$k = \exp(500.0/T)$
24	$\text{BrO} + \text{HO}_2 \longrightarrow \text{O}_2 + \text{HOBr}$	$k = (3.4\text{e-}12) \exp(540.0/T)$
25	$\text{BrO} + \text{NO}_3 \longrightarrow$ $\text{Br} + \text{O}_2 + \text{NO}_2$	$k = (1.0\text{e-}12)$

26	$\text{BrO} + \text{CH}_3\text{O}_2 \longrightarrow$ $\text{O}_2 + \text{Br} + \text{CH}_3\text{O}$	$k = (3.23\text{e-}11) \exp(-332.0/T)$
27	$\text{CH}_3 + \text{H}_2\text{O} \longrightarrow \text{OH} + \text{CH}_4$	$k = (1.2\text{e-}14) \exp(-62190.0/T)(T/298.0)^{2.9}$
28	$\text{CH}_3 + \text{HCl} \longrightarrow \text{Cl} + \text{CH}_4$	$k = (3.88\text{e-}13) \exp(-9640.0/T)$
29	$\text{CH}_4 + \text{H}_2\text{O} \longrightarrow \text{CH}_3 + \text{H}_2\text{O}_2$	$k = (3.0\text{e-}13) \exp(-77740.0/T)$
30	$\text{CH}_4 + \text{HCO} \longrightarrow \text{CH}_3 + \text{HCHO}$	$k = (1.36\text{e-}13) \exp(-94200.0/T)(T/298.0)^{2.85}$
31	$\text{CH}_4 + \text{CH}_3\text{O} \longrightarrow \text{CH}_3 + \text{CH}_3\text{OH}$	$k = (2.6\text{e-}13) \exp(-37000.0/T)$

32	$\text{CH}_4 + \text{CH}_3\text{O}_2 \longrightarrow \text{CH}_3 + \text{CH}_3\text{OOH}$	$k = (3.0\text{e-}13) \exp(-77320.0/T)$
33	$\text{CO} + \text{HO}_2 \longrightarrow \text{OH} + \text{CO}_2$	$k = (2.5\text{e-}10) \exp(-98940.0/T)$
34	$\text{CO} + \text{CH}_3\text{O} \longrightarrow \text{CH}_3 + \text{CO}_2$	$k = (2.6\text{e-}11) \exp(-49390.0/T)$
35	$\text{CO} + \text{NO}_2 \longrightarrow \text{NO} + \text{CO}_2$	$k = (1.48\text{e-}10) \exp(-141420.0/T)$
36	$\text{CO} + \text{NO}_3 \longrightarrow \text{NO}_2 + \text{CO}_2$	$k = (4.0\text{e-}19)$
37	$\text{Cl} + \text{H}_2 \longrightarrow \text{H} + \text{HCl}$	$k = (3.7\text{e-}11) \exp(-2300.0/T)$
38	$\text{Cl} + \text{O}_3 \longrightarrow \text{O}_2 + \text{ClO}$	$k = (2.9\text{e-}11) \exp(-260.0/T)$

39	$\text{Cl} + \text{CH}_4 \longrightarrow \text{HCl} + \text{CH}_3$	$k = (1.1\text{e-}11) \exp(-1400.0/T)$
40	$\text{Cl} + \text{H}_2\text{O} \longrightarrow \text{OH} + \text{HCl}$	$k = (2.79\text{e-}11) \exp(-72080.0/T)$
41	$\text{Cl} + \text{HCl} \longrightarrow \text{H} + \text{Cl}_2$	$k = (1.66\text{e-}7) \exp(-198590.0/T)$
42	$\text{Cl} + \text{HO}_2 \longrightarrow \text{O}_2 + \text{HCl}$	$k = (1.8\text{e-}11) \exp(170.0/T)$
43	$\text{Cl} + \text{HO}_2 \longrightarrow \text{OH} + \text{ClO}$	$k = (4.1\text{e-}11) \exp(-450.0/T)$
44	$\text{Cl} + \text{N}_2\text{O} \longrightarrow \text{N}_2 + \text{ClO}$	$k = (2.16\text{e-}10) \exp(-140150.0/T)$
45	$\text{Cl} + \text{NO}_3 \longrightarrow \text{ClO} + \text{NO}_2$	$k = (2.4\text{e-}11)$

46	$\text{Cl} + \text{ClOO} \longrightarrow \text{ClO} + \text{ClO}$	$k = (1.2\text{e-}11)$
47	$\text{Cl} + \text{ClOO} \longrightarrow \text{O}_2 + \text{Cl}_2$	$k = (2.3\text{e-}10)$
48	$\text{Cl} + \text{H}_2\text{O}_2 \longrightarrow \text{HCl} + \text{HO}_2$	$k = (1.1\text{e-}11) \exp(-980.0/T)$
49	$\text{Cl} + \text{HCHO} \longrightarrow \text{HCl} + \text{HCO}$	$k = (8.2\text{e-}11) \exp(-34.0/T)$
50	$\text{Cl} + \text{HOCl} \longrightarrow \text{OH} + \text{Cl}_2$	$k = (2.5\text{e-}12) \exp(-130.0/T)$
51	$\text{Cl} + \text{CH}_3\text{O}_2 \longrightarrow \text{CH}_3\text{O} + \text{ClO}$	$k = (7.7\text{e-}11)$
52	$\text{Cl} + \text{OClO} \longrightarrow \text{ClO} + \text{ClO}$	$k = (3.4\text{e-}11) \exp(160.0/T)$

53	$\text{Cl} + \text{Cl}_2\text{O}_2 \longrightarrow \text{Cl}_2 + \text{ClOO}$	$k = (1.0\text{e-}10)$
54	$\text{Cl} + \text{CH}_3\text{OCl} \longrightarrow$ $\text{Cl}_2 + \text{HO}_2 + \text{HCHO}$	$k = (4.9\text{e-}11)$
55	$\text{Cl} + \text{CH}_3\text{OOH} \longrightarrow \text{HCl} + \text{CH}_3\text{O}_2$	$k = (5.9\text{e-}11)$
56	$\text{Cl} + \text{BrONO}_2 \longrightarrow \text{NO}_3 + \text{BrCl}$	$k = (2.0\text{e-}11) \exp(329.0/T)$
57	$\text{Cl} + \text{ClONO}_2 \longrightarrow \text{Cl}_2 + \text{NO}_3$	$k = (6.5\text{e-}12) \exp(135.0/T)$
58	$\text{Cl} + \text{CH}_3\text{ONO}_2 \longrightarrow$ $\text{NO}_2 + \text{HCl} + \text{HCHO}$	$k = (1.3\text{e-}11) \exp(-1200.0/T)$
59	$\text{Cl} + \text{CH}_3\text{O}_2\text{NO}_2 \longrightarrow$ $\text{NO}_3 + \text{HCl} + \text{HCHO}$	$k = (4.0\text{e-}13) \exp(-640.0/T)$

60	$\text{ClO} + \text{ClO} \longrightarrow \text{Cl} + \text{ClOO}$	$k = (3.0\text{e-}11) \exp(-2450.0/T)$
61	$\text{ClO} + \text{ClO} \longrightarrow \text{O}_2 + \text{Cl}_2$	$k = (1.0\text{e-}12) \exp(-1590.0/T)$
62	$\text{ClO} + \text{ClO} \longrightarrow \text{Cl} + \text{OClO}$	$k = (3.5\text{e-}13) \exp(-1370.0/T)$
63	$\text{ClO} + \text{HO}_2 \longrightarrow \text{O}_2 + \text{HOCl}$	$k = (4.8\text{e-}13) \exp(700.0/T)$
64	$\text{ClO} + \text{HO}_2 \longrightarrow \text{O}_3 + \text{HCl}$	$k = \exp(710.0/T)$
65	$\text{ClO} + \text{NO}_3 \longrightarrow \text{NO}_2 + \text{ClOO}$	$k = (4.7\text{e-}13)$
66	$\text{ClO} + \text{NO}_3 \longrightarrow \text{NO}_2 + \text{OClO}$	$k = 0$

67	$\text{ClO} + \text{HCHO} \longrightarrow \text{HCO} + \text{HOCl}$	$k = (1.0\text{e-}15)$
68	$\text{ClO} + \text{CH}_3\text{O}_2 \longrightarrow \text{O}_2 + \text{CH}_3\text{OCl}$	$k = (2.6\text{e-}13) \exp(263.0/T)$
69	$\text{ClO} + \text{CH}_3\text{O}_2 \longrightarrow \text{CH}_3\text{O} + \text{ClOO}$	$k = (4.9\text{e-}12) \exp(-330.0/T)$
70	$\begin{array}{c} \text{ClO} + \text{CH}_3\text{O}_2 \longrightarrow \\ \text{O}_2 + \text{Cl} + \text{CH}_3\text{O} \end{array}$	$k = (4.91\text{e-}12) \exp(-332.0/T)$
71	$\text{H} + \text{O}_3 \longrightarrow \text{OH} + \text{O}_2$	$k = (1.4\text{e-}10) \exp(-480.0/T)$
72	$\text{H} + \text{Br}_2 \longrightarrow \text{Br} + \text{HBr}$	$k = (1.84\text{e-}10) \exp(-558.0/T)(T/298.0)^{0.5}$

73	$\text{H} + \text{CH}_4 \longrightarrow \text{H}_2 + \text{CH}_3$	$k = (5.82\text{e-}13) \exp(-33630.0/T)(T/298.0)^{3.0}$
74	$\text{H} + \text{H}_2\text{O} \longrightarrow \text{OH} + \text{H}_2$	$k = (6.83\text{e-}12) \exp(-80810.0/T)(T/298.0)^{1.6}$
75	$\text{H} + \text{HCl} \longrightarrow \text{H}_2 + \text{Cl}$	$k = (1.32\text{e-}11) \exp(-14220.0/T)$
76	$\text{H} + \text{HO}_2 \longrightarrow \text{O}({}^3\text{P}) + \text{H}_2\text{O}$	$k = (2.4\text{e-}12)$
77	$\text{H} + \text{HO}_2 \longrightarrow \text{OH} + \text{OH}$	$k = (7.2\text{e-}11)$
78	$\text{H} + \text{HO}_2 \longrightarrow \text{H}_2 + \text{O}_2$	$k = (5.6\text{e-}12)$

79	$\text{H} + \text{N}_2\text{O} \longrightarrow \text{N}_2 + \text{OH}$	$k = (9.793\text{e-}11) \exp(-56640.0/T)$
80	$\text{H} + \text{NO}_2 \longrightarrow \text{OH} + \text{NO}$	$k = (4.0\text{e-}10) \exp(-340.0/T)$
81	$\text{H} + \text{HONO} \longrightarrow \text{H}_2 + \text{NO}_2$	$k = (2.0\text{e-}11) \exp(-3700.0/T)$
82	$\text{H} + \text{CH}_3\text{Br} \longrightarrow \text{CH}_3 + \text{HBr}$	$k = (4.34\text{e-}11) \exp(-2646.0/T)$
83	$\text{H}_2 + \text{Br}_2 \longrightarrow \text{HBr} + \text{HBr}$	$k = (6.81\text{e-}9) \exp(-20430.0/T)$
84	$\text{H}_2\text{O} + \text{N}_2\text{O}_5 \longrightarrow \text{HNO}_3 + \text{HNO}_3$	$k = (2.0\text{e-}21)$

85	$\text{H}_2\text{O} + \text{ClONO}_2 \longrightarrow \text{HOCl} + \text{HNO}_3$	$k = (2.0\text{e-}21)$
86	$\text{HCO} + \text{H}_2\text{O} \longrightarrow \text{OH} + \text{HCHO}$	$k = (8.54\text{e-}13) \exp(-109300.0/T)(T/298.0)^{1.35}$
87	$\text{HCO} + \text{HONO} \longrightarrow \text{NO}_2 + \text{HCHO}$	$k = (2.0\text{e-}21)T^{2.37} \exp(-1940.0/T)$
88	$\text{HCl} + \text{NO}_3 \longrightarrow \text{Cl} + \text{HNO}_3$	$k = (5.0\text{e-}17)$
89	$\text{HCl} + \text{N}_2\text{O}_5 \longrightarrow \text{HNO}_3 + \text{ClNO}_2$	$k = (6.97\text{e-}21)$
90	$\text{HCl} + \text{ClONO}_2 \longrightarrow \text{Cl}_2 + \text{HNO}_3$	$k = (1.0\text{e-}20)$

91	$\text{HCl} + \text{HO}_2\text{NO}_2 \longrightarrow \text{HOCl} + \text{HNO}_3$	$k = (1.0\text{e-}21)$
92	$\text{HO}_2 + \text{HO}_2 \longrightarrow \text{O}_2 + \text{H}_2\text{O}_2$	$k = (2.2\text{e-}13) \exp(600.0/T)(1.0+(0.6*P/1013.25))$
93	$\text{HO}_2 + \text{NO}_3 \longrightarrow$ $\text{O}_2 + \text{OH} + \text{NO}_2$	$k = (2.15\text{e-}12)$
94	$\text{HO}_2 + \text{NO}_3 \longrightarrow \text{O}_2 + \text{HNO}_3$	$k = (2.15\text{e-}12)$
95	$\text{HO}_2 + \text{HCHO} \longrightarrow \text{HCO} + \text{H}_2\text{O}_2$	$k = (3.3\text{e-}12) \exp(-48800.0/T)$
96	$\text{HO}_2 + \text{CH}_3\text{O}_2 \longrightarrow \text{O}_2 + \text{CH}_3\text{OOH}$	$k = (3.42\text{e-}13) \exp(780.0/T)$

97	$\text{HO}_2 + \text{CH}_3\text{O}_2 \longrightarrow$ $\text{O}_2 + \text{H}_2\text{O} + \text{HCHO}$	$k = (3.42\text{e-}14) \exp(780.0/T)$
98	$\text{CH}_3\text{O}_2 + \text{CH}_3\text{O}_2 \longrightarrow$ $\text{O}_2 + \text{CH}_3\text{O} + \text{CH}_3\text{O}$	$k = (3.67\text{e-}14) \exp(365.0/T)$
99	$\text{CH}_3\text{O}_2 + \text{CH}_3\text{O}_2 \longrightarrow$ $\text{O}_2 + \text{CH}_3\text{OH} + \text{HCHO}$	$k = (6.6\text{e-}14) \exp(365.0/T)$
100	$\text{N} + \text{NO} \longrightarrow \text{O}({}^3\text{P}) + \text{N}_2$	$k = (3.1\text{e-}11)$
101	$\text{N} + \text{O}_2 \longrightarrow \text{O}({}^3\text{P}) + \text{NO}$	$k = (4.4\text{e-}12) \exp(-3220.0/T)$
102	$\text{N} + \text{O}_3 \longrightarrow \text{NO} + \text{O}_2$	$k = (1.0\text{e-}16)$
103	$\text{N} + \text{OH} \longrightarrow \text{H} + \text{NO}$	$k = (3.8\text{e-}11) \exp(85.0/T)$

104	$\text{N} + \text{NO}_2 \longrightarrow \text{O}({}^3\text{P}) + \text{N}_2\text{O}$	$k = (3.0\text{e-}12)$
105	$\text{NO} + \text{O}_3 \longrightarrow \text{O}_2 + \text{NO}_2$	$k = (1.8\text{e-}12) \exp(-1370.0/T)$
106	$\text{NO} + \text{BrO} \longrightarrow \text{Br} + \text{NO}_2$	$k = (8.8\text{e-}12) \exp(260.0/T)$
107	$\text{NO} + \text{ClO} \longrightarrow \text{Cl} + \text{NO}_2$	$k = (6.4\text{e-}12) \exp(290.0/T)$
108	$\text{NO} + \text{HO}_2 \longrightarrow \text{OH} + \text{NO}_2$	$k = (3.5\text{e-}12) \exp(250.0/T)$
109	$\text{NO} + \text{N}_2\text{O} \longrightarrow \text{N}_2 + \text{NO}_2$	$k = (2.51\text{e-}13) \exp(-206270.0/T)$
110	$\text{NO} + \text{NO}_3 \longrightarrow \text{NO}_2 + \text{NO}_2$	$k = (1.5\text{e-}11) \exp(170.0/T)$

111	$\text{NO} + \text{CH}_3\text{O}_2 \longrightarrow \text{CH}_3\text{O} + \text{NO}_2$	$k = (4.1\text{e-}12) \exp(180.0/T)$
112	$\text{NO} + \text{CH}_3\text{O}_2 \longrightarrow \text{CH}_3\text{ONO}_2$	$k = (4.1\text{e-}15) \exp(180.0/T)$
113	$\text{NO} + \text{OClO} \longrightarrow \text{NO}_2 + \text{ClO}$	$k = (3.4\text{e-}13)$
114	$\text{NO} + \text{HNO}_3 \longrightarrow \text{NO}_2 + \text{HONO}$	$k = (7.43\text{e-}21)$
115	$\text{NO}_2 + \text{NO}_3 \longrightarrow$ $\text{O}_2 + \text{NO} + \text{NO}_2$	$k = (4.5\text{e-}14) \exp(-1260.0/T)$
116	$\text{NO}_3 + \text{HBr} \longrightarrow \text{Br} + \text{HNO}_3$	$k = (1.0\text{e-}16)$
117	$\text{NO}_3 + \text{HCHO} \longrightarrow \text{HCO} + \text{HNO}_3$	$k = (5.8\text{e-}16)$

118	$\text{NO}_3 + \text{CH}_3\text{OH} \longrightarrow \text{CH}_3\text{O} + \text{HNO}_3$	$k = (1.3\text{e-}12) \exp(-2560.0/T)$
119	$\text{NO}_3 + \text{CH}_3\text{O}_2 \longrightarrow$ $\text{O}_2 + \text{CH}_3\text{O} + \text{NO}_2$	$k = (1.0\text{e-}12)$
120	$\text{NO}_3 + \text{OCIO} \longrightarrow$ $\text{O}_2 + \text{ClO} + \text{NO}_2$	$k = (2.0\text{e-}15)$
121	$\text{O}(^1\text{D}) + \text{CO} \longrightarrow \text{CO}_2$	$k = (7.3\text{e-}11)$
122	$\text{O}(^1\text{D}) + \text{H}_2 \longrightarrow \text{H} + \text{OH}$	$k = (1.1\text{e-}10)$
123	$\text{O}(^1\text{D}) + \text{N}_2 \longrightarrow \text{O}(^3\text{P}) + \text{N}_2$	$k = (1.8\text{e-}11) \exp(107.0/T)$
124	$\text{O}(^1\text{D}) + \text{NO} \longrightarrow \text{N} + \text{O}_2$	$k = (5.0\text{e-}11)$

125	$\text{O}(^1\text{D}) + \text{O}_2 \longrightarrow \text{O}(^3\text{P}) + \text{O}_2$	$k = (3.2\text{e-}11) \exp(67.0/T)$
126	$\text{O}(^1\text{D}) + \text{O}_3 \longrightarrow$ $\text{O}(^3\text{P}) + \text{O}(^3\text{P}) + \text{O}_2$	$k = (1.2\text{e-}10)$
127	$\text{O}(^1\text{D}) + \text{O}_3 \longrightarrow \text{O}_2 + \text{O}_2$	$k = (1.2\text{e-}10)$
128	$\text{O}(^1\text{D}) + \text{CH}_4 \longrightarrow \text{OH} + \text{CH}_3$	$k = (1.13\text{e-}10)$
129	$\text{O}(^1\text{D}) + \text{CH}_4 \longrightarrow \text{H}_2 + \text{HCHO}$	$k = (7.5\text{e-}12)$
130	$\text{O}(^1\text{D}) + \text{CH}_4 \longrightarrow$ $\text{H} + \text{H} + \text{HCHO}$	$k = (3.0\text{e-}11)$
131	$\text{O}(^1\text{D}) + \text{CO}_2 \longrightarrow \text{O}(^3\text{P}) + \text{CO}_2$	$k = (7.4\text{e-}11) \exp(120.0/T)$

132	$\text{O}(^1\text{D}) + \text{Cl}_2 \longrightarrow \text{O}(^3\text{P}) + \text{Cl}_2$	$k = (7.0\text{e-}11)$
133	$\text{O}(^1\text{D}) + \text{Cl}_2 \longrightarrow \text{Cl} + \text{ClO}$	$k = (2.1\text{e-}10)$
134	$\text{O}(^1\text{D}) + \text{H}_2\text{O} \longrightarrow \text{OH} + \text{OH}$	$k = (2.2\text{e-}10)$
135	$\text{O}(^1\text{D}) + \text{HBr} \longrightarrow \text{O}(^3\text{P}) + \text{HBr}$	$k = (3.0\text{e-}11)$
136	$\text{O}(^1\text{D}) + \text{HBr} \longrightarrow \text{OH} + \text{Br}$	$k = (1.5\text{e-}10)$
137	$\text{O}(^1\text{D}) + \text{HCl} \longrightarrow \text{O}(^3\text{P}) + \text{HCl}$	$k = (1.35\text{e-}11)$
138	$\text{O}(^1\text{D}) + \text{HCl} \longrightarrow \text{H} + \text{ClO}$	$k = (3.6\text{e-}11)$

139	$\text{O}(^1\text{D}) + \text{HCl} \longrightarrow \text{OH} + \text{Cl}$	$k = (1.0\text{e-}10)$
140	$\text{O}(^1\text{D}) + \text{N}_2\text{O} \longrightarrow \text{N} + \text{NO}_2$	$k = (2.45\text{e-}13)$
141	$\text{O}(^1\text{D}) + \text{N}_2\text{O} \longrightarrow \text{NO} + \text{NO}$	$k = (7.2\text{e-}11)$
142	$\text{O}(^1\text{D}) + \text{N}_2\text{O} \longrightarrow \text{N}_2 + \text{O}_2$	$k = (4.4\text{e-}11)$
143	$\text{O}(^1\text{D}) + \text{CF}_2\text{Cl}_2 \longrightarrow \text{Cl} + \text{Cl}$	$k = (1.4\text{e-}10)$
144	$\text{O}(^3\text{P}) + \text{H}_2 \longrightarrow \text{H} + \text{OH}$	$k = (9.0\text{e-}18)$
145	$\text{O}(^3\text{P}) + \text{O}_3 \longrightarrow \text{O}_2 + \text{O}_2$	$k = (8.0\text{e-}12) \exp(-2060.0/T)$

146	$\text{O}(^3\text{P}) + \text{OH} \longrightarrow \text{H} + \text{O}_2$	$k = (2.3\text{e-}11) \exp(110.0/T)$
147	$\text{O}(^3\text{P}) + \text{Br}_2 \longrightarrow \text{Br} + \text{BrO}$	$k = (1.4\text{e-}11)$
148	$\text{O}(^3\text{P}) + \text{BrO} \longrightarrow \text{O}_2 + \text{Br}$	$k = (1.9\text{e-}11) \exp(230.0/T)$
149	$\text{O}(^3\text{P}) + \text{CH}_3 \longrightarrow \text{H} + \text{HCHO}$	$k = (1.4\text{e-}10)$
150	$\text{O}(^3\text{P}) + \text{CH}_4 \longrightarrow \text{OH} + \text{CH}_3$	$k = (8.32\text{e-}12) \exp(-35500.0/T)(T/298.0)^{1.56}$
151	$\text{O}(^3\text{P}) + \text{ClO} \longrightarrow \text{Cl} + \text{O}_2$	$k = (3.0\text{e-}11) \exp(70.0/T)$
152	$\text{O}(^3\text{P}) + \text{H}_2\text{O} \longrightarrow \text{OH} + \text{OH}$	$k = (1.85\text{e-}11) \exp(-71260.0/T)(T/298.0)^{0.946}$

153	$\text{O}(^3\text{P}) + \text{HBr} \longrightarrow \text{OH} + \text{Br}$	$k = (6.6\text{e-}12) \exp(-1540.0/T)$
154	$\text{O}(^3\text{P}) + \text{HCl} \longrightarrow \text{OH} + \text{Cl}$	$k = (1.0\text{e-}11) \exp(-3340.0/T)$
155	$\text{O}(^3\text{P}) + \text{HO}_2 \longrightarrow \text{OH} + \text{O}_2$	$k = (2.7\text{e-}11) \exp(224.0/T)$
156	$\text{O}(^3\text{P}) + \text{NO}_2 \longrightarrow \text{O}_2 + \text{NO}$	$k = (6.5\text{e-}12) \exp(120.0/T)$
157	$\text{O}(^3\text{P}) + \text{NO}_3 \longrightarrow \text{O}_2 + \text{NO}_2$	$k = (1.7\text{e-}11)$
158	$\text{O}(^3\text{P}) + \text{BrCl} \longrightarrow \text{Cl} + \text{BrO}$	$k = (2.2\text{e-}11)$
159	$\text{O}(^3\text{P}) + \text{H}_2\text{O}_2 \longrightarrow \text{OH} + \text{HO}_2$	$k = (1.4\text{e-}12) \exp(-2000.0/T)$

160	$\text{O}(^3\text{P}) + \text{HCHO} \longrightarrow \text{OH} + \text{HCO}$	$k = (3.4\text{e-}11) \exp(-1600.0/T)$
161	$\text{O}(^3\text{P}) + \text{HOBr} \longrightarrow \text{OH} + \text{BrO}$	$k = (1.4\text{e-}10) \exp(-430.0/T)$
162	$\text{O}(^3\text{P}) + \text{HOCl} \longrightarrow \text{OH} + \text{ClO}$	$k = (1.7\text{e-}13)$
163	$\text{O}(^3\text{P}) + \text{HONO} \longrightarrow \text{OH} + \text{NO}_2$	$k = (2.0\text{e-}11) \exp(-3000.0/T)$
164	$\text{O}(^3\text{P}) + \text{OCIO} \longrightarrow \text{O}_2 + \text{ClO}$	$k = (2.5\text{e-}12) \exp(-950.0/T)$
165	$\text{O}(^3\text{P}) + \text{CH}_3\text{Br} \longrightarrow \text{OH} + \text{Br}$	$k = (7.6\text{e-}13) \exp(-890.0/T)$
166	$\text{O}(^3\text{P}) + \text{HNO}_3 \longrightarrow \text{OH} + \text{NO}_3$	$k = (3.0\text{e-}17)$

167	$\text{O}(^3\text{P}) + \text{ClONO}_2 \longrightarrow \text{NO}_2 + \text{OCIO}$	$k = (1.5\text{e-}12) \exp(-800.0/T)$
168	$\text{O}(^3\text{P}) + \text{ClONO}_2 \longrightarrow \text{NO}_3 + \text{ClO}$	$k = (1.5\text{e-}12) \exp(-800.0/T)$
169	$\text{O}(^3\text{P}) + \text{HO}_2\text{NO}_2 \longrightarrow$ $\text{OH} + \text{O}_2 + \text{NO}_2$	$k = (7.8\text{e-}11) \exp(-3400.0/T)$
170	$\text{O}_2 + \text{CH}_3 \longrightarrow \text{OH} + \text{HCHO}$	$k = (3.0\text{e-}16)$
171	$\text{O}_2 + \text{CH}_4 \longrightarrow \text{CH}_3 + \text{HO}_2$	$k = (6.7\text{e-}11) \exp(-238120.0/T)(T/298.0)^{1.56}$
172	$\text{O}_2 + \text{HCO} \longrightarrow \text{CO} + \text{HO}_2$	$k = (5.5\text{e-}12)$

173	$\text{O}_2 + \text{CH}_3\text{O} \longrightarrow \text{HO}_2 + \text{HCHO}$	$k = (7.2\text{e-}14) \exp(-1080.0/T)$
174	$\text{O}_3 + \text{BrO} \longrightarrow$ $\text{Br} + \text{O}_2 + \text{O}_2$	$k = (5.0\text{e-}15)$
175	$\text{O}_3 + \text{CH}_3 \longrightarrow \text{O}_2 + \text{CH}_3\text{O}$	$k = (5.1\text{e-}12) \exp(-210.0/T)$
176	$\text{O}_3 + \text{CH}_3 \longrightarrow$ $\text{H} + \text{O}_2 + \text{HCHO}$	$k = (5.1\text{e-}12) \exp(-210.0/T)$
177	$\text{O}_3 + \text{ClO} \longrightarrow \text{O}_2 + \text{ClOO}$	$k = (1.5\text{e-}17)$
178	$\text{O}_3 + \text{ClO} \longrightarrow \text{O}_2 + \text{OClO}$	$k = (1.0\text{e-}18) \exp(-4000.0/T)$
179	$\text{O}_3 + \text{HO}_2 \longrightarrow$ $\text{OH} + \text{O}_2 + \text{O}_2$	$k = (1.4\text{e-}14) \exp(-600.0/T)$

180	$\text{O}_3 + \text{NO}_2 \longrightarrow \text{O}_2 + \text{NO}_3$	$k = (1.2\text{e-}13) \exp(-2450.0/T)$
181	$\text{O}_3 + \text{HONO} \longrightarrow \text{O}_2 + \text{HNO}_3$	$k = (5.0\text{e-}19)$
182	$\text{O}_3 + \text{Cl}_2\text{O}_2 \longrightarrow$ $\text{O}_2 + \text{ClO} + \text{ClOO}$	$k = (1.0\text{e-}19)$
183	$\text{OH} + \text{CO} \longrightarrow \text{H} + \text{CO}_2$	$k = (1.5\text{e-}13)(1.0 + (0.6 * P/1013.25))$
184	$\text{OH} + \text{H}_2 \longrightarrow \text{H} + \text{H}_2\text{O}$	$k = (5.0\text{e-}12) \exp(-2000.0/T)$
185	$\text{OH} + \text{O}_3 \longrightarrow \text{O}_2 + \text{HO}_2$	$k = (1.9\text{e-}12) \exp(-1000.0/T)$
186	$\text{OH} + \text{OH} \longrightarrow \text{O}({}^3\text{P}) + \text{H}_2\text{O}$	$k = (4.2\text{e-}12) \exp(-240.0/T)$

187	$\text{OH} + \text{Br}_2 \longrightarrow \text{Br} + \text{HOBr}$	$k = (1.2\text{e-}11) \exp(400.0/T)$
188	$\text{OH} + \text{BrO} \longrightarrow \text{Br} + \text{HO}_2$	$k = (7.425\text{e-}11)$
189	$\text{OH} + \text{BrO} \longrightarrow \text{O}_2 + \text{HBr}$	$k = (7.5\text{e-}13)$
190	$\text{OH} + \text{CH}_4 \longrightarrow \text{H}_2\text{O} + \text{CH}_3$	$k = (2.8\text{e-}14)T^{0.667} \exp(-1575.0/T)$
191	$\text{OH} + \text{Cl}_2 \longrightarrow \text{Cl} + \text{HOCl}$	$k = (1.4\text{e-}12) \exp(-900.0/T)$
192	$\text{OH} + \text{ClO} \longrightarrow \text{Cl} + \text{HO}_2$	$k = (9.35\text{e-}12) \exp(120.0/T)$
193	$\text{OH} + \text{ClO} \longrightarrow \text{O}_2 + \text{HCl}$	$k = (1.65\text{e-}12) \exp(120.0/T)$

194	$\text{OH} + \text{HBr} \longrightarrow \text{Br} + \text{H}_2\text{O}$	$k = (1.1\text{e-}11)$
195	$\text{OH} + \text{HCl} \longrightarrow \text{Cl} + \text{H}_2\text{O}$	$k = (2.6\text{e-}12) \exp(-350.0/T)$
196	$\text{OH} + \text{HO}_2 \longrightarrow \text{O}_2 + \text{H}_2\text{O}$	$k = (4.8\text{e-}11) \exp(250.0/T)$
197	$\text{OH} + \text{N}_2\text{O} \longrightarrow \text{N}_2 + \text{HO}_2$	$k = (3.8\text{e-}17)$
198	$\text{OH} + \text{NO}_3 \longrightarrow \text{HO}_2 + \text{NO}_2$	$k = (2.3\text{e-}11)$
199	$\text{OH} + \text{H}_2\text{O}_2 \longrightarrow \text{H}_2\text{O} + \text{HO}_2$	$k = (2.9\text{e-}12) \exp(-160.0/T)$
200	$\text{OH} + \text{HCHO} \longrightarrow \text{H}_2\text{O} + \text{HCO}$	$k = (8.8\text{e-}12) \exp(25.0/T)$

201	$\text{OH} + \text{HOCl} \longrightarrow \text{ClO} + \text{H}_2\text{O}$	$k = (3.0\text{e-}12) \exp(-500.0/T)$
202	$\text{OH} + \text{HONO} \longrightarrow \text{H}_2\text{O} + \text{NO}_2$	$k = (1.8\text{e-}11) \exp(-390.0/T)$
203	$\text{OH} + \text{CH}_3\text{OH} \longrightarrow \text{H}_2\text{O} + \text{CH}_3\text{O}$	$k = (5.0\text{e-}13) \exp(-380.0/T)$
204	$\text{OH} + \text{OClO} \longrightarrow \text{O}_2 + \text{HOCl}$	$k = (4.5\text{e-}13) \exp(800.0/T)$
205	$\text{OH} + \text{ClNO}_2 \longrightarrow \text{NO}_2 + \text{HOCl}$	$k = (3.5\text{e-}14)$
206	$\text{OH} + \text{HNO}_3 \longrightarrow \text{H}_2\text{O} + \text{NO}_3$	$k = 2.4\text{e-}14 \cdot \exp(460.0/T) + \frac{6.50\text{e-}34 \cdot M \cdot \exp(1335.0/T)}{1 + \frac{6.50\text{e-}34 \cdot M \cdot \exp(1335.0/T)}{2.7\text{e-}17 \cdot \exp(2199.0/T)}}$
207	$\text{OH} + \text{CH}_3\text{OCl} \longrightarrow$ $\text{Cl} + \text{H}_2\text{O} + \text{HCHO}$	$k = (2.4\text{e-}12) \exp(-360.0/T)$

208	$\text{OH} + \text{CH}_3\text{OOH} \longrightarrow \text{H}_2\text{O} + \text{CH}_3\text{O}_2$	$k = (1.9\text{e-}12) \exp(190.0/T)$
209	$\text{OH} + \text{CH}_3\text{OOH} \longrightarrow$ $\text{OH} + \text{H}_2\text{O} + \text{HCHO}$	$k = (1.0\text{e-}12) \exp(190.0/T)$
210	$\text{OH} + \text{ClONO}_2 \longrightarrow \text{NO}_3 + \text{HOCl}$	$k = (1.2\text{e-}12) \exp(-330.0/T)$
211	$\text{OH} + \text{HO}_2\text{NO}_2 \longrightarrow$ $\text{O}_2 + \text{H}_2\text{O} + \text{NO}_2$	$k = (1.5\text{e-}12) \exp(360.0/T)$
212	$\text{OH} + \text{CH}_3\text{ONO}_2 \longrightarrow$ $\text{NO}_2 + \text{H}_2\text{O} + \text{HCHO}$	$k = (4.0\text{e-}13) \exp(-845.0/T)$
213	$\text{OH} + \text{CH}_3\text{O}_2\text{NO}_2 \longrightarrow$ $\text{NO}_3 + \text{H}_2\text{O} + \text{HCHO}$	$k = (1.0\text{e-}13) \exp(-640.0/T)$

214	$\text{N}_2^+ + \text{O}(^3\text{P}) \longrightarrow \text{NO}^+ + \text{N}$	$k = (1.4\text{e-}10)$
215	$\text{N}_2^+ + \text{O}_2 \longrightarrow \text{O}_2^+ + \text{N}_2$	$k = (6.0\text{e-}11)$
216	$\text{N}_2^+ + \text{NO} \longrightarrow \text{NO}^+ + \text{N}_2$	$k = (5.0\text{e-}10)$
217	$\text{N}_2^+ + \text{CO}_2 \longrightarrow \text{CO}_2^+ + \text{N}_2$	$k = (7.7\text{e-}10)$
218	$\text{N}_2^+ + \text{CO} \longrightarrow \text{CO}^+ + \text{N}_2$	$k = (7.4\text{e-}11)$
219	$\text{N}^+ + \text{O}_2 \longrightarrow \text{O}_2^+ + \text{N}$	$k = (2.8\text{e-}10)$
220	$\text{N}^+ + \text{NO} \longrightarrow \text{NO}^+ + \text{N}$	$k = (8.0\text{e-}10)$

221	$\text{N}^+ + \text{CO}_2 \longrightarrow \text{CO}_2^+ + \text{N}$	$k = (7.5\text{e-}10)$
222	$\text{N}^+ + \text{CO}_2 \longrightarrow \text{CO}^+ + \text{NO}$	$k = (2.5\text{e-}10)$
223	$\text{O}^+ + \text{N}_2 \longrightarrow \text{NO}^+ + \text{N}$	$k = (1.2\text{e-}12)$
224	$\text{O}^+ + \text{O}_2 \longrightarrow \text{O}_2^+ + \text{O}(^3\text{P})$	$k = (1.2\text{e-}12)$
225	$\text{O}^+ + \text{NO} \longrightarrow \text{NO}^+ + \text{O}(^3\text{P})$	$k = (1.1\text{e-}11)$
226	$\text{O}^+ + \text{CO}_2 \longrightarrow \text{CO}_2^+ + \text{O}(^3\text{P})$	$k = (1.1\text{e-}9)$
227	$\text{O}^+ + \text{CO}_2 \longrightarrow \text{O}_2^+ + \text{CO}$	$k = (1.1\text{e-}9)$

228	$\text{O}_2^+ + \text{NO} \longrightarrow \text{NO}^+ + \text{O}_2$	$k = (4.4\text{e-}10)$
229	$\text{Ar}^+ + \text{N}_2 \longrightarrow \text{N}_2^+ + \text{Ar}$	$k = (2.5\text{e-}12)$
230	$\text{Ar}^+ + \text{CO}_2 \longrightarrow \text{CO}_2^+ + \text{Ar}$	$k = (4.2\text{e-}10)$
231	$\text{Ar}^+ + \text{O}_2 \longrightarrow \text{O}_2^+ + \text{Ar}$	$k = (5.5\text{e-}11)$
232	$\text{Ar}^+ + \text{NO} \longrightarrow \text{NO}^+ + \text{Ar}$	$k = (2.0\text{e-}10)$
233	$\text{CO}_2^+ + \text{O}_2 \longrightarrow \text{O}_2^+(\text{CO}_2)$	$k = (1.0\text{e-}10)$
234	$\text{CO}_2^+ + \text{NO} \longrightarrow \text{NO}^+ + \text{CO}_2$	$k = (2.0\text{e-}11)$

235	$\text{CO}_2^+ + \text{CO} \longrightarrow \text{CO}^+ + \text{CO}_2$	$k = (1.9\text{e-}12)$
236	$\text{CO}^+ + \text{CO}_2 \longrightarrow \text{CO}_2^+ + \text{CO}$	$k = (1.42\text{e-}9)$
237	$\text{CO}_2^+ \text{CO}_2 + \text{O}_2 \longrightarrow \text{O}_2^+ + 2\text{CO}_2$	$k = (1.0\text{e-}10)$
238	$\text{CO}_2^+ \text{CO}_2 + \text{O}_2 \longrightarrow$ $\text{O}_2^+(\text{CO}_2) + \text{CO}_2$	$k = (2.7\text{e-}11)$
239	$\text{O}_2^+(\text{CO}_2) + \text{CO}_2 \longrightarrow \text{O}_2^+ + 2\text{CO}_2$	$k = (2.4\text{e-}13)$
240	$\text{O}_2^+(\text{CO}_2) + \text{H}_2\text{O} \longrightarrow$ $\text{O}_2^+(\text{H}_2\text{O}) + \text{CO}_2$	$k = (1.1\text{e-}9)$
241	$\text{O}_2^+(\text{CO}_2)_2 + \text{H}_2\text{O} \longrightarrow$ $\text{O}_2^+(\text{H}_2\text{O}) + 2\text{CO}_2$	$k = (2.3\text{e-}9)$

242	$\text{O}_2^+(\text{H}_2\text{O}) + \text{H}_2\text{O} \longrightarrow$ $\text{H}_3\text{O}^+ + \text{OH} + \text{O}_2$	$k = (2.04\text{e-}10)$
243	$\text{O}_2^+(\text{H}_2\text{O}) + \text{H}_2\text{O} \longrightarrow \text{H}_3\text{O}^+\text{OH} + \text{O}_2$	$k = (9.96\text{e-}10)$
244	$\text{O}_2^+(\text{H}_2\text{O})_2 + \text{H}_2\text{O} \longrightarrow$ $\text{O}_2^+(\text{H}_2\text{O}) + 2\text{H}_2\text{O}$	$k = (3.24\text{e-}10)$
245	$\text{O}_2^+\text{O}_2 + \text{CO}_2 \longrightarrow \text{O}_2^+(\text{CO}_2) + \text{O}_2$	$k = (4.0\text{e-}13)$
246	$\text{O}_2^+\text{O}_2 + \text{H}_2\text{O} \longrightarrow \text{O}_2^+(\text{H}_2\text{O}) + \text{O}_2$	$k = (1.7\text{e-}9)$
247	$\text{H}_3\text{O}^+\text{OH} + \text{H}_2\text{O} \longrightarrow \text{H}_3\text{O}^+(\text{H}_2\text{O}) + \text{OH}$	$k = (1.4\text{e-}9)$

248	$\text{O}(^3\text{P}) + \text{e}^- \longrightarrow \text{O}^-$	$k = (1.3\text{e-}15)$
249	$\text{O}_3 + \text{e}^- \longrightarrow \text{O}^- + \text{O}_2$	$k = (9.1\text{e-}12) \exp(-46.0/T)(T/300.0)^{-1.0}$
250	$\text{O}_3^- + \text{CO}_2 \longrightarrow \text{CO}_3^- + \text{O}_2$	$k = (5.5\text{e-}10)$
251	$\text{CO}_3^- + \text{NO} \longrightarrow \text{NO}_2^- + \text{CO}_2$	$k = (1.0\text{e-}11)$
252	$\text{CO}_3^- + \text{NO}_2 \longrightarrow \text{NO}_3^- + \text{CO}_2$	$k = (2.0\text{e-}10)$
253	$\text{CO}_4^- + \text{NO} \longrightarrow \text{NO}_3^- + \text{CO}_2$	$k = (4.8\text{e-}11)$
254	$\text{CO}_4^- + \text{O}(^3\text{P}) \longrightarrow \text{CO}_3^- + \text{O}_2$	$k = (1.4\text{e-}10)$

255	$\text{CO}_4^- + \text{O}_3 \longrightarrow$ $\text{O}_3^- + \text{CO}_2 + \text{O}_2$	$k = (1.3\text{e-}10)$
256	$\text{NO}_2^- + \text{O}_3 \longrightarrow \text{NO}_3^- + \text{O}_2$	$k = (1.2\text{e-}10)$
257	$\text{NO}_3^- + \text{CO}_2 \longrightarrow \text{CO}_3^- + \text{NO}_2$	$k = (1.0\text{e-}11)$
258	$\text{O}_2^+\text{O}_2 + h\nu \longrightarrow \text{O}_2^+ + \text{O}_2$	$k = (0.13)$
259	$\text{O}_2^+(\text{H}_2\text{O}) + h\nu \longrightarrow \text{O}_2^+ + \text{H}_2\text{O}$	$k = (0.18)$
260	$\text{O}_2^+(\text{H}_2\text{O})_2 + h\nu \longrightarrow \text{O}_2^+(\text{H}_2\text{O}) + \text{H}_2\text{O}$	$k = (0.27)$
261	$\text{CO}_3^- + h\nu \longrightarrow$ $\text{e}^- + \text{CO}_2 + \text{O}(^3\text{P})$	$k = (0.01)$

262	$\text{CO}_3^- + h\nu \longrightarrow \text{e}^- + \text{CO}_2$	$k = (0.076)$
263	$\text{CO}_4^- + h\nu \longrightarrow \text{O}_2^- + \text{CO}_2$	$k = (0.0028)$
264	$\text{NO}_2^- + h\nu \longrightarrow \text{e}^- + \text{NO}_2$	$k = (0.018)$
265	$\text{NO}_3^- + h\nu \longrightarrow \text{O}({}^3\text{P}) + \text{NO}_2^-$	$k = (0.026)$
266	$\text{NO}_3^- + h\nu \longrightarrow$ $\text{e}^- + \text{NO} + \text{O}_2$	$k = (0.002)$
267	$\text{O}^- + h\nu \longrightarrow \text{e}^- + \text{O}({}^3\text{P})$	$k = (0.62)$
268	$\text{O}_2^- + h\nu \longrightarrow \text{e}^- + \text{O}_2$	$k = (0.17)$

269	$\text{O}_3^- + h\nu \longrightarrow \text{e}^- + \text{O}_3$	$k = (0.021)$
270	$\text{O}_3^- + h\nu \longrightarrow \text{O}^- + \text{O}_2$	$k = (0.21)$
271	$\text{N}_2^+ + \text{e}^- \longrightarrow 2\text{N}$	$k = (2.7\text{e-}7)$
272	$\text{N}_2^+ + \text{O}^- \longrightarrow \text{N}_2 + \text{O}(^3\text{P})$	$k = (1.5\text{e-}7)$
273	$\text{N}_2^+ + \text{O}_2^- \longrightarrow \text{N}_2 + \text{O}_2$	$k = (1.5\text{e-}7)$
274	$\text{N}_2^+ + \text{O}_3^- \longrightarrow \text{N}_2 + \text{O}_3$	$k = (1.5\text{e-}7)$
275	$\text{N}_2^+ + \text{NO}_2^- \longrightarrow \text{N}_2 + \text{NO}_2$	$k = (1.5\text{e-}7)$

276	$\text{N}_2^+ + \text{NO}_3^- \longrightarrow \text{N}_2 + \text{NO}_3$	$k = (1.7\text{e-}7)$
277	$\text{N}_2^+ + \text{CO}_3^- \longrightarrow$ $\text{N}_2 + \text{CO}_2 + \text{O}(^3\text{P})$	$k = (1.5\text{e-}7)$
278	$\text{N}_2^+ + \text{NO}_2^-(\text{H}_2\text{O})_2 \longrightarrow$ $\text{N}_2 + \text{NO}_2 + 2\text{H}_2\text{O}$	$k = (1.5\text{e-}7)$
279	$\text{N}_2^+ + \text{CO}_4^- \longrightarrow$ $\text{N}_2 + \text{CO}_2 + \text{O}_2$	$k = (1.5\text{e-}7)$
280	$\text{O}_2^+ + \text{e}^- \longrightarrow 2\text{O}(^3\text{P})$	$k = (2.8\text{e-}7)(T/300.0)^{-0.63}$
281	$\text{O}_2^+ + \text{O}^- \longrightarrow \text{O}_2 + \text{O}(^3\text{P})$	$k = (1.0\text{e-}7)(T/300.0)^{-0.5}$
282	$\text{O}_2^+ + \text{O}_2^- \longrightarrow 2\text{O}_2$	$k = (4.2\text{e-}7)(T/300.0)^{-0.5}$

283	$\text{O}_2^+ + \text{O}_3^- \longrightarrow \text{O}_2 + \text{O}_3$	$k = (4.0\text{e-}7)(T/300.0)^{-0.5}$
284	$\text{O}_2^+ + \text{NO}_2^- \longrightarrow \text{O}_2 + \text{NO}_2$	$k = (4.1\text{e-}7)(T/300.0)^{-0.5}$
285	$\text{O}_2^+ + \text{NO}_3^- \longrightarrow \text{O}_2 + \text{NO}_3$	$k = (1.3\text{e-}7)(T/300.0)^{-0.5}$
286	$\text{O}_2^+ + \text{CO}_3^- \longrightarrow$ $\text{O}_2 + \text{CO}_2 + \text{O}(^3\text{P})$	$k = (2.0\text{e-}7)(T/300.0)^{-0.5}$
287	$\text{O}_2^+ + \text{NO}_2^-(\text{H}_2\text{O})_2 \longrightarrow$ $\text{O}_2 + \text{NO}_2 + 2\text{H}_2\text{O}$	$k = (2.0\text{e-}7)(T/300.0)^{-0.5}$
288	$\text{O}_2^+ + \text{CO}_4^- \longrightarrow 2\text{O}_2 + 2\text{CO}_2$	$k = (2.0\text{e-}7)$
289	$\text{NO}^+ + \text{e}^- \longrightarrow \text{N} + \text{O}(^3\text{P})$	$k = (4.0\text{e-}7)(T/300.0)^{-1.0}$

290	$\text{NO}^+ + \text{O}^- \longrightarrow \text{NO} + \text{O}(^3\text{P})$	$k = (4.9\text{e-}7)(T/300.0)^{-0.5}$
291	$\text{NO}^+ + \text{O}_2^- \longrightarrow \text{NO} + \text{O}_2$	$k = (6.0\text{e-}7)(T/300.0)^{-0.5}$
292	$\text{NO}^+ + \text{O}_3^- \longrightarrow \text{NO} + \text{O}_3$	$k = (1.0\text{e-}7)(T/300.0)^{-0.5}$
293	$\text{NO}^+ + \text{NO}_2^- \longrightarrow \text{NO} + \text{NO}_2$	$k = (1.0\text{e-}7)(T/300.0)^{-0.5}$
294	$\text{NO}^+ + \text{NO}_3^- \longrightarrow \text{NO}_2 + \text{NO}_3$	$k = (1.3\text{e-}7)(T/300.0)^{-0.5}$
295	$\text{NO}^+ + \text{CO}_3^- \longrightarrow$ $\text{NO} + \text{CO}_2 + \text{O}(^3\text{P})$	$k = (1.2\text{e-}7)(T/300.0)^{-0.5}$
296	$\text{NO}^+ + \text{NO}_2^-(\text{H}_2\text{O})_2 \longrightarrow$ $\text{NO} + \text{NO}_2 + 2\text{H}_2\text{O}$	$k = (1.0\text{e-}7)(T/300.0)^{-0.5}$

297	$\text{NO}^+ + \text{CO}_4^- \longrightarrow$ $\text{NO} + \text{CO}_2 + \text{O}_2$	$k = (1.0\text{e-}7)(T/300.0)^{-0.5}$
298	$\text{CO}_2^+ + \text{e}^- \longrightarrow \text{CO} + \text{O}(^3\text{P})$	$k = (2.7\text{e-}7)$
299	$\text{CO}_2^+ + \text{O}^- \longrightarrow \text{CO}_2 + \text{O}(^3\text{P})$	$k = (1.0\text{e-}7)(T/300.0)^{-0.5}$
300	$\text{CO}_2^+ + \text{O}_2^- \longrightarrow \text{CO}_2 + \text{O}_2$	$k = (2.0\text{e-}7)(T/300.0)^{-0.5}$
301	$\text{CO}_2^+ + \text{O}_3^- \longrightarrow \text{CO}_2 + \text{O}_3$	$k = (2.0\text{e-}7)(T/300.0)^{-0.5}$
302	$\text{CO}_2^+ + \text{NO}_2^- \longrightarrow \text{CO}_2 + \text{NO}_2$	$k = (2.0\text{e-}7)(T/300.0)^{-0.5}$
303	$\text{CO}_2^+ + \text{NO}_3^- \longrightarrow \text{CO}_2 + \text{NO}_3$	$k = (2.0\text{e-}7)(T/300.0)^{-0.5}$

304	$\text{CO}_2^+ + \text{CO}_3^- \longrightarrow 2\text{CO}_2 + \text{O}(^3\text{P})$	$k = (2.0\text{e-}7)(T/300.0)^{-0.5}$
305	$\text{CO}_2^+ + \text{NO}_2^-(\text{H}_2\text{O})_2 \longrightarrow$ $\text{CO}_2 + \text{NO}_2 + 2\text{H}_2\text{O}$	$k = (2.0\text{e-}7)(T/300.0)^{-0.5}$
306	$\text{CO}_2^+ + \text{CO}_4^- \longrightarrow 2\text{CO}_2 + \text{O}_2$	$k = (2.0\text{e-}7)(T/300.0)^{-0.5}$
307	$\text{H}_3\text{O}^+ + \text{e}^- \longrightarrow \text{H}_2 + \text{OH}$	$k = (1.3\text{e-}6)(T/300.0)^{-0.7}$
308	$\text{H}_3\text{O}^+ + \text{O}^- \longrightarrow$ $\text{H}_2 + \text{O}(^3\text{P}) + \text{OH}$	$k = (2.0\text{e-}7)(T/300.0)^{-0.5}$
309	$\text{H}_3\text{O}^+ + \text{O}_2^- \longrightarrow$ $\text{H}_2 + \text{O}_2 + \text{OH}$	$k = (2.0\text{e-}7)(T/300.0)^{-0.5}$
310	$\text{H}_3\text{O}^+ + \text{O}_3^- \longrightarrow$ $\text{H}_2 + \text{O}_3 + \text{OH}$	$k = (2.0\text{e-}7)(T/300.0)^{-0.5}$

311	$\text{H}_3\text{O}^+ + \text{NO}_2^- \longrightarrow$ $\text{H}_2 + \text{NO}_2 + \text{OH}$	$k = (2.0\text{e-}7)(T/300.0)^{-0.5}$
312	$\text{H}_3\text{O}^+ + \text{NO}_3^- \longrightarrow$ $\text{H}_2 + \text{NO}_3 + \text{OH}$	$k = (2.0\text{e-}7)(T/300.0)^{-0.5}$
313	$\text{H}_3\text{O}^+ + \text{CO}_3^- \longrightarrow$ $\text{H}_2 + \text{CO}_2 + \text{O}({}^3\text{P}) + \text{OH}$	$k = (2.0\text{e-}7)(T/300.0)^{-0.5}$
314	$\text{H}_3\text{O}^+ + \text{NO}_2^- (\text{H}_2\text{O})_2 \longrightarrow$ $\text{H}_2 + \text{NO}_2 + 2\text{H}_2\text{O} + \text{OH}$	$k = (2.0\text{e-}7)(T/300.0)^{-0.5}$
315	$\text{H}_3\text{O}^+ + \text{CO}_4^- \longrightarrow$ $\text{H}_2 + \text{CO}_2 + \text{O}_2 + \text{OH}$	$k = (2.0\text{e-}7)(T/300.0)^{-0.5}$
316	$\text{H}_3\text{O}^+(\text{H}_2\text{O}) + \text{e}^- \longrightarrow$ $\text{H}_2 + \text{OH} + \text{H}_2\text{O}$	$k = (2.8\text{e-}6)(T/300.0)^{-0.15}$
317	$\text{H}_3\text{O}^+(\text{H}_2\text{O}) + \text{O}_2^- \longrightarrow$ $\text{H}_2 + \text{O}_2 + \text{OH} + \text{H}_2\text{O}$	$k = (2.0\text{e-}7)(T/300.0)^{-0.5}$

318	$\text{H}_3\text{O}^+(\text{H}_2\text{O}) + \text{O}_3^- \longrightarrow$ $\text{H}_2 + \text{O}_3 + \text{OH} + \text{H}_2\text{O}$	$k = (2.0\text{e-}7)(T/300.0)^{-0.5}$
319	$\text{H}_3\text{O}^+(\text{H}_2\text{O}) + \text{NO}_2^- \longrightarrow$ $\text{H}_2 + \text{NO}_2 + \text{OH} + \text{H}_2\text{O}$	$k = (2.0\text{e-}7)(T/300.0)^{-0.5}$
320	$\text{H}_3\text{O}^+(\text{H}_2\text{O}) + \text{NO}_3^- \longrightarrow$ $\text{H}_2 + \text{NO}_3 + \text{OH} + \text{H}_2\text{O}$	$k = (2.0\text{e-}7)(T/300.0)^{-0.5}$
321	$\text{H}_3\text{O}^+(\text{H}_2\text{O}) + \text{CO}_3^- \longrightarrow$ $\text{H}_2 + \text{CO}_2 + \text{O}({}^3\text{P}) + \text{OH} + \text{H}_2\text{O}$	$k = (2.0\text{e-}7)(T/300.0)^{-0.5}$
322	$\text{H}_3\text{O}^+(\text{H}_2\text{O}) + \text{NO}_2^-(\text{H}_2\text{O})_2 \longrightarrow$ $\text{H}_2 + \text{NO}_2 + \text{OH} + 3\text{H}_2\text{O}$	$k = (2.0\text{e-}7)(T/300.0)^{-0.5}$
323	$\text{CO}_2^+\text{CO}_2 + \text{e}^- \longrightarrow 2\text{CO}_2$	$k = (2.0\text{e-}6)(T/300.0)^{-1.0}$
324	$\text{O}_2^+\text{O}_2 + \text{e}^- \longrightarrow 2\text{O}_2$	$k = (2.0\text{e-}6)(T/300.0)^{-1.0}$

325	$\text{O}_2^+\text{O}_2 + \text{O}^- \longrightarrow 2\text{O}_2 + \text{O}(^3\text{P})$	$k = (2.0\text{e-}7)(T/300.0)^{-0.5}$
326	$\text{O}_2^+\text{O}_2 + \text{O}_2^- \longrightarrow 3\text{O}_2$	$k = (2.0\text{e-}7)(T/300.0)^{-0.5}$
327	$\text{O}_2^+\text{O}_2 + \text{O}_3^- \longrightarrow 2\text{O}_2 + \text{O}_3$	$k = (2.0\text{e-}7)(T/300.0)^{-0.5}$
328	$\text{O}_2^+\text{O}_2 + \text{NO}_3^- \longrightarrow 2\text{O}_2 + \text{NO}_3$	$k = (2.0\text{e-}7)(T/300.0)^{-0.5}$
329	$\text{O}_2^+\text{O}_2 + \text{CO}_3^- \longrightarrow$ $2\text{O}_2 + \text{CO}_2 + \text{O}(^3\text{P})$	$k = (2.0\text{e-}7)(T/300.0)^{-0.5}$
330	$\text{O}_2^+\text{O}_2 + \text{NO}_2^-(\text{H}_2\text{O})_2 \longrightarrow$ $2\text{O}_2 + \text{NO}_2 + 2\text{H}_2\text{O}$	$k = (2.0\text{e-}7)(T/300.0)^{-0.5}$
331	$\text{O}_2^+\text{O}_2 + \text{CO}_4^- \longrightarrow 3\text{O}_2 + \text{CO}_2$	$k = (2.0\text{e-}7)(T/300.0)^{-0.5}$

332	$\text{O}_2^- + \text{HNO}_3 \longrightarrow \text{NO}_3^- + \text{HO}_2$	$k = (2.8\text{e-}9)$
333	$\text{O}_2^- + \text{HCl} \longrightarrow \text{Cl}^- + \text{HO}_2$	$k = (1.6\text{e-}9)$
334	$\text{O}^- + \text{O}_3 \longrightarrow \text{O}_3^- + \text{O}(^3\text{P})$	$k = (8.0\text{e-}10)$
335	$\text{O}^- + \text{NO}_2 \longrightarrow \text{NO}_2^- + \text{O}(^3\text{P})$	$k = (1.0\text{e-}9)$
336	$\text{O}^- + \text{H}_2 \longrightarrow \text{OH}^- + \text{H}$	$k = (3.2\text{e-}11)$
337	$\text{O}^- + \text{CH}_4 \longrightarrow \text{OH}^- + \text{CH}_3$	$k = (1.0\text{e-}10)$
338	$\text{O}^- + \text{H}_2 \longrightarrow \text{H}_2\text{O} + \text{e}^-$	$k = (6.0\text{e-}10)$

339	$\text{O}^- + \text{O}(^3\text{P}) \longrightarrow \text{O}_2 + \text{e}^-$	$k = (1.9\text{e-}10)$
340	$\text{O}^- + \text{NO} \longrightarrow \text{NO}_2 + \text{e}^-$	$k = (2.8\text{e-}10)$
341	$\text{O}^- + \text{HCl} \longrightarrow \text{Cl}^- + \text{OH}$	$k = (2.0\text{e-}9)$
342	$\text{O}^- + \text{HNO}_3 \longrightarrow \text{NO}_3^- + \text{OH}$	$k = (3.0\text{e-}9)$
343	$\text{O}_2^- + \text{O}(^3\text{P}) \longrightarrow \text{O}_2 + \text{O}^-$	$k = (1.5\text{e-}10)$
344	$\text{O}_2^- + \text{O}_3 \longrightarrow \text{O}_2 + \text{O}_3^-$	$k = (7.8\text{e-}10)$
345	$\text{O}_2^- + \text{NO}_2 \longrightarrow \text{NO}_2^- + \text{O}_2$	$k = (7.8\text{e-}10)$

346	$\text{O}_2^- + \text{O}(^3\text{P}) \longrightarrow \text{e}^- + \text{O}_3$	$k = (1.5\text{e-}10)$
347	$\text{O}_2^- + \text{H} \longrightarrow \text{e}^- + \text{HO}_2$	$k = (1.5\text{e-}10)$
348	$\text{O}_2^-(\text{H}_2\text{O}) + \text{CO}_2 \longrightarrow \text{CO}_4^- + \text{HO}_2$	$k = (5.8\text{e-}10)$
349	$\text{O}_3^- + \text{O}(^3\text{P}) \longrightarrow \text{O}_2^- + \text{O}_2$	$k = (2.5\text{e-}10)$
350	$\text{O}_3^- + \text{H} \longrightarrow \text{OH}^- + \text{O}_2$	$k = (8.4\text{e-}10)$
351	$\text{O}_3^- + \text{NO}_2 \longrightarrow \text{NO}_3^- + \text{O}_2$	$k = (2.8\text{e-}10)$
352	$\text{O}_3^- + \text{NO} \longrightarrow \text{NO}_3^- + \text{O}(^3\text{P})$	$k = (4.5\text{e-}12)$

353	$\text{CO}_3^- + \text{N}_2\text{O}_5 \longrightarrow$ $\text{NO}_3^- + \text{CO}_2 + \text{NO}_3$	$k = (2.8\text{e-}10)$
354	$\text{CO}_3^- + \text{O}(^3\text{P}) \longrightarrow \text{O}_2^- + \text{CO}_2$	$k = (1.1\text{e-}10)$
355	$\text{CO}_3^- + \text{H} \longrightarrow \text{OH}^- + \text{CO}_2$	$k = (1.7\text{e-}10)$
356	$\text{NO}_2^- + \text{NO}_2 \longrightarrow \text{NO}_3^- + \text{NO}$	$k = (2.0\text{e-}13)$
357	$\text{NO}_2^- + \text{H} \longrightarrow \text{OH}^- + \text{NO}$	$k = (3.0\text{e-}10)$
358	$\text{NO}_2^- + \text{HCl} \longrightarrow \text{Cl}^- + \text{HONO}$	$k = (1.4\text{e-}9)$
359	$\text{NO}_2^- + \text{HNO}_3 \longrightarrow \text{NO}_3^- + \text{HONO}$	$k = (1.6\text{e-}9)$

360	$\text{NO}_2^- + \text{N}_2\text{O}_5 \longrightarrow \text{NO}_3^- + 2\text{NO}_2$	$k = (7.0\text{e-}10)$
361	$\text{OH}^- + \text{NO}_2 \longrightarrow \text{NO}_2^- + 2\text{OH}$	$k = (1.1\text{e-}9)$
362	$\text{OH}^- + \text{O}_3 \longrightarrow \text{O}_3^- + 2\text{OH}$	$k = (9.0\text{e-}10)$
363	$\text{OH}^- + \text{H} \longrightarrow \text{e}^- + 2\text{H}_2\text{O}$	$k = (1.4\text{e-}9)$
364	$\text{OH}^- + \text{O}(^3\text{P}) \longrightarrow \text{e}^- + 2\text{HO}_2$	$k = (2.0\text{e-}10)$
365	$\text{O}_4^- + \text{O}(^3\text{P}) \longrightarrow \text{O}_3^- + 2\text{O}_2$	$k = (4.0\text{e-}10)$
366	$\text{O}_4^- + \text{CO}_2 \longrightarrow \text{CO}_4^- + 2\text{O}_2$	$k = (4.3\text{e-}10)$

367	$\text{O}_4^- + \text{NO} \longrightarrow \text{NO}_3^- + 2\text{O}_2$	$k = (2.5\text{e-}10)$
368	$\text{O}_4^- + \text{H}_2\text{O} \longrightarrow \text{O}_2^-(\text{H}_2\text{O}) + 2\text{O}_2$	$k = (1.0\text{e-}10)$
369	$\text{CO}_4^- + \text{H} \longrightarrow \text{CO}_3^- + 2\text{OH}$	$k = (2.2\text{e-}10)$
370	$\text{CO}_4^- + \text{H}_2\text{O} \longrightarrow \text{O}_2^-(\text{H}_2\text{O}) + 2\text{CO}_2$	$k = (2.5\text{e-}10)$
371	$\text{Cl}^- + \text{H} \longrightarrow \text{HCl} + 2\text{e}^-$	$k = (9.3\text{e-}10)$
372	$\text{Cl}^- + \text{HNO}_3 \longrightarrow \text{HCl} + 2\text{NO}_3^-$	$k = (1.6\text{e-}9)$
373	$\text{Cl}^- + \text{N}_2\text{O}_5 \longrightarrow \text{NO}_3^- + 2\text{ClNO}_2$	$k = (9.4\text{e-}10)$

A.1.2 Trimolecular Reactions

#	Trimolecular Reaction	Reaction Rate Coeff
1	$\text{Br} + \text{NO}_2 \longrightarrow \text{BrONO} + \text{M}$	$k_0 = M(4.2\text{e-}31)(T/300.0)^{-2.4}$ $k_\infty = (2.7\text{e-}11)(T/300.0)^{0.0}$ $f_c = (0.55)$
2	$\text{BrO} + \text{NO}_2 \longrightarrow \text{BrONO}_2 + \text{M}$	$k_0 = M(5.2\text{e-}31)(T/300.0)^{-3.2}$ $k_\infty = (6.9\text{e-}12)(T/300.0)^{-2.9}$ $f_c = (0.6)$
3	$\text{CH}_3 + \text{O}_2 \longrightarrow \text{CH}_3\text{O}_2 + \text{M}$	$k_0 = M(1.0\text{e-}30)(T/300.0)^{-3.3}$ $k_\infty = (2.2\text{e-}12)(T/300.0)^{1.0}$ $f_c = (0.27)$
4	$\text{CO} + \text{H} \longrightarrow \text{HCO} + \text{M}$	$k_0 = M(1.9\text{e-}33) \exp(-7000.0/T)(T/300.0)^{0.0}$ $k_\infty = (T/298.0)^{0.0}$ $f_c = (0.0)$

5	$\text{CO} + \text{O}(^3\text{P}) \longrightarrow \text{CO}_2 + \text{M}$	$k_0 = M(1.7\text{e-}33) \exp(-12550.0/T)(T/300.0)^{0.0}$ $k_\infty = (4.2\text{e-}12) \exp(-199540.0/T)(T/298.0)^{0.0}$ $f_c = (0.6)$
6	$\text{Cl}_2\text{O}_2 + \text{M} \longrightarrow \text{ClO} + \text{ClO}$	$k_0 = M(1.35\text{e-}5) \exp(-8720.0/T)(T/300.0)^{1.0}$ $k_\infty = (1.8\text{e}15) \exp(-8450.0/T)(T/300.0)^{0.0}$ $f_c = (0.6)$
7	$\text{Cl} + \text{NO}_2 \longrightarrow \text{ClNO}_2 + \text{M}$	$k_0 = M(1.8\text{e-}31)(T/300.0)^{-2.0}$ $k_\infty = (1.0\text{e-}10)(T/300.0)^{-1.0}$ $f_c = (0.6)$
8	$\text{Cl} + \text{O}_2 \longrightarrow \text{ClOO} + \text{M}$	$k_0 = M(2.7\text{e-}33)(T/300.0)^{-1.5}$ $k_\infty = (T/300.0)^{0.0}$ $f_c = (0.0)$
9	$\text{ClO} + \text{ClO} \longrightarrow \text{Cl}_2\text{O}_2 + \text{M}$	$k_0 = M(2.2\text{e-}32)(T/300.0)^{-3.1}$ $k_\infty = (3.5\text{e-}12)(T/300.0)^{-1.0}$ $f_c = (0.6)$

10	$\text{ClO} + \text{NO}_2 \longrightarrow \text{ClONO}_2 + \text{M}$	$k_0 = M(1.6\text{e-}31)(T/300.0)^{-3.4}$ $k_\infty = (2.0\text{e-}11)(T/300.0)^{0.0}$ $f_c = (0.0) + (0.0) \cdot \exp(-T/430.0)$
11	$\text{ClONO}_2 + \text{M} \longrightarrow \text{ClO} + \text{NO}_2$	$k_0 = M(T/300.0)^{0.0}$ $k_\infty = (2.75\text{e-}6) \exp(-95100.0/T)(T/300.0)^{0.0}$ $f_c = (0.0)$
12	$\text{ClOO} + \text{M} \longrightarrow \text{Cl} + \text{O}_2$	$k_0 = M(2.8\text{e-}10) \exp(-1820.0/T)(T/300.0)^{0.0}$ $k_\infty = (T/300.0)^{0.0}$ $f_c = (0.0)$
13	$\text{HO}_2 + \text{HO}_2 \longrightarrow \text{H}_2\text{O}_2 + \text{O}_2$	$k_0 = M(1.9\text{e-}33) \exp(980.0/T)(T/300.0)^{0.0}$ $k_\infty = (T/300.0)^{0.0}$ $f_c = (0.0)$

14	$\text{H} + \text{O}_2 \longrightarrow \text{HO}_2 + \text{M}$	$k_0 = M(6.2\text{e-}32)(T/300.0)^{-1.6}$ $k_\infty = (7.5\text{e-}11)(T/300.0)^{0.0}$ $f_c = (0.0) + (0.0) \cdot \exp(-T/498.0)$
15	$\text{HO}_2 + \text{NO}_2 \longrightarrow \text{HO}_2\text{NO}_2 + \text{M}$	$k_0 = M(1.8\text{e-}31)(T/300.0)^{-3.2}$ $k_\infty = (4.7\text{e-}12)(T/300.0)^{-1.4}$ $f_c = (0.6)$
16	$\text{HO}_2\text{NO}_2 + \text{M} \longrightarrow \text{HO}_2 + \text{NO}_2$	$k_0 = M(5.0\text{e-}6) \exp(-10000.0/T)(T/300.0)^{0.0}$ $k_\infty = (2.6\text{e}15) \exp(-10900.0/T)(T/300.0)^{0.0}$ $f_c = (0.6)$
17	$\text{HONO} + \text{M} \longrightarrow \text{NO} + \text{OH}$	$k_0 = M(6.4\text{e}6)T^{-3.8} \exp(-25340.0/T)$ $k_\infty = (1.2\text{e}19)T^{-1.23} \exp(-25010.0/T)$ $f_c = (0.62)$

18	$\text{CH}_3\text{O}_2\text{NO}_2 + \text{M} \longrightarrow \text{CH}_3\text{O}_2 + \text{NO}_2$	$k_0 = M(9.0\text{e-}5) \exp(-9690.0/T)(T/300.0)^{0.0}$ $k_\infty = (1.1\text{e}16) \exp(-10560.0/T)(T/300.0)^{0.0}$ $f_c = (0.4)$
19	$\text{CH}_3\text{O} + \text{NO}_2 \longrightarrow \text{CH}_3\text{ONO}_2 + \text{M}$	$k_0 = M(2.8\text{e-}29)(T/300.0)^{-4.5}$ $k_\infty = (2.0\text{e-}11)(T/300.0)^{0.0}$ $f_c = (0.44)$
20	$\text{CH}_3\text{O}_2 + \text{NO}_2 \longrightarrow \text{CH}_3\text{O}_2\text{NO}_2 + \text{M}$	$k_0 = M(2.5\text{e-}30)(T/300.0)^{-5.5}$ $k_\infty = (7.5\text{e-}12)(T/300.0)^{0.0}$ $f_c = (0.4)$
21	$\text{N}_2 + \text{O}(^1\text{D}) \longrightarrow \text{N}_2\text{O} + \text{M}$	$k_0 = M(3.5\text{e-}37)(T/300.0)^{-0.6}$ $k_\infty = (T/300.0)^{0.0}$ $f_c = (0.0)$
22	$\text{N}_2\text{O}_5 + \text{M} \longrightarrow \text{NO}_2 + \text{NO}_3$	$k_0 = M(0.0022) \exp(-11080.0/T)(T/300.0)^{-4.4}$ $k_\infty = (9.7\text{e}14) \exp(-11080.0/T)(T/300.0)^{0.1}$ $f_c = (0.0) + (0.0) \cdot \exp(-T/250.0) + \exp(\text{extrm{-1}}$

23	$\text{NO}_2 + \text{NO}_3 \longrightarrow \text{N}_2\text{O}_5 + \text{M}$	$k_0 = M(2.7\text{e-}30)(T/300.0)^{-3.4}$ $k_\infty = (2.0\text{e-}12)(T/300.0)^{0.2}$ $f_c = (0.0) + (0.0) \cdot \exp(-T/250.0) + \exp(\text{extrm{-1}}$
24	$\text{NO}_2 + \text{O}(^3\text{P}) \longrightarrow \text{NO}_3 + \text{M}$	$k_0 = M(9.0\text{e-}32)(T/300.0)^{-2.0}$ $k_\infty = (2.2\text{e-}11)(T/300.0)^{0.0}$ $f_c = (0.0) + (0.0) \cdot \exp(-T/1300.0)$
25	$\text{NO}_2 + \text{OH} \longrightarrow \text{HNO}_3 + \text{M}$	$k_0 = M(2.47\text{e-}30)(T/300.0)^{-2.97}$ $k_\infty = (1.45\text{e-}11)(T/300.0)^{-2.77}$ $f_c = (0.6)$
26	$\text{NO} + \text{NO} \longrightarrow \text{NO}_2 + \text{NO}_2$	$k_0 = M \exp(530.0/T)(T/300.0)^{0.0}$ $k_\infty = (T/300.0)^{0.0}$ $f_c = (0.0)$

27	$\text{NO} + \text{O}(^3\text{P}) \longrightarrow \text{NO}_2 + \text{M}$	$k_0 = M(1.0\text{e-}31)(T/300.0)^{-1.6}$ $k_\infty = (3.0\text{e-}11)(T/300.0)^{0.3}$ $f_c = (0.0) + (0.0) \cdot \exp(-T/1850.0)$
28	$\text{NO} + \text{OH} \longrightarrow \text{HONO} + \text{M}$	$k_0 = M(7.4\text{e-}31)(T/300.0)^{-2.4}$ $k_\infty = (3.2\text{e-}11)(T/300.0)^{0.0}$ $f_c = (0.0) + (0.0) \cdot \exp(-T/1300.0)$
29	$\text{O}(^3\text{P}) + \text{O}_2 \longrightarrow \text{O}_3 + \text{M}$	$k_0 = M(5.64\text{e-}34)(T/300.0)^{-2.8}$ $k_\infty = (T/300.0)^{0.0}$ $f_c = (0.0)$
30	$\text{OH} + \text{OH} \longrightarrow \text{H}_2\text{O}_2 + \text{M}$	$k_0 = M(8.0\text{e-}31)(T/300.0)^{-0.8}$ $k_\infty = (3.0\text{e-}11)(T/300.0)^{0.0}$ $f_c = (0.5)$

31	$\text{CH}_3\text{ONO}_2 + \text{M} \longrightarrow \text{CH}_3\text{O} + \text{NO}_2$	$k_0 = M(T/300.0)^{0.0}$ $k_\infty = (1.0\text{e}13) \exp(-140.0/T)(T/300.0)^{0.0}$ $f_c = (0.0)$
32	$\text{O}_2^+ + \text{O}_2 \longrightarrow \text{O}_2^+\text{O}_2 + \text{M}$	$k_0 = M(1.0\text{e-}30)(T/300.0)^{0.0}$ $k_\infty = (T/300.0)^{0.0}$ $f_c = (0.0)$
33	$\text{O}_2^+ + \text{H}_2\text{O} \longrightarrow \text{O}_2^+(\text{H}_2\text{O}) + \text{M}$	$k_0 = M(2.8\text{e-}28)(T/300.0)^{0.0}$ $k_\infty = (T/300.0)^{0.0}$ $f_c = (0.0)$
34	$\text{O}_2^+ + \text{CO}_2 \longrightarrow \text{O}_2^+(\text{CO}_2) + \text{M}$	$k_0 = M(1.7\text{e-}29)(T/300.0)^{0.0}$ $k_\infty = (T/300.0)^{0.0}$ $f_c = (0.0)$
35	$\text{CO}_2^+ + \text{CO}_2 \longrightarrow \text{CO}_2^+\text{CO}_2 + \text{M}$	$k_0 = M(2.5\text{e-}28)(T/300.0)^{0.0}$ $k_\infty = (T/300.0)^{0.0}$ $f_c = (0.0)$

36	$\text{O}_2^+(\text{CO}_2) + \text{CO}_2 \longrightarrow$ $\text{O}_2^+(\text{CO}_2)_2 + \text{M}$	$k_0 = M(5.0\text{e-}30)(T/300.0)^{0.0}$ $k_\infty = (T/300.0)^{0.0}$ $f_c = (0.0)$
37	$\text{O}_2^+(\text{H}_2\text{O}) + \text{H}_2\text{O} \longrightarrow$ $\text{O}_2^+(\text{H}_2\text{O})_2 + \text{M}$	$k_0 = M(1.3\text{e-}27)(T/300.0)^{0.0}$ $k_\infty = (T/300.0)^{0.0}$ $f_c = (0.0)$
38	$\text{H}_3\text{O}^+ + \text{H}_2\text{O} \longrightarrow \text{H}_3\text{O}^+(\text{H}_2\text{O}) + \text{M}$	$k_0 = M(3.4\text{e-}27)(T/300.0)^{0.0}$ $k_\infty = (T/300.0)^{0.0}$ $f_c = (0.0)$
39	$\text{H}_3\text{O}^+(\text{H}_2\text{O}) + \text{H}_2\text{O} \longrightarrow$ $\text{H}_3\text{O}^+(\text{H}_2\text{O})_2 + \text{M}$	$k_0 = M(2.3\text{e-}27)(T/300.0)^{0.0}$ $k_\infty = (T/300.0)^{0.0}$ $f_c = (0.0)$
40	$\text{H}_3\text{O}^+(\text{H}_2\text{O}) + \text{M} \longrightarrow$ $\text{H}_3\text{O}^+ + \text{H}_2\text{O} + \text{M}$	$k_0 = M(7.0\text{e-}26)(T/300.0)^{0.0}$ $k_\infty = (T/300.0)^{0.0}$ $f_c = (0.0)$

41	$\text{H}_3\text{O}^+(\text{H}_2\text{O})_2 + \text{H}_2\text{O} \longrightarrow$ $\text{H}_3\text{O}^+(\text{H}_2\text{O})_3 + \text{M}$	$k_0 = M(2.4\text{e-}27)(T/300.0)^{0.0}$ $k_\infty = (T/300.0)^{0.0}$ $f_c = (0.0)$
42	$\text{H}_3\text{O}^+(\text{H}_2\text{O})_2 + \text{M} \longrightarrow$ $\text{H}_3\text{O}^+(\text{H}_2\text{O}) + \text{H}_2\text{O} + \text{M}$	$k_0 = M(7.0\text{e-}18)(T/300.0)^{0.0}$ $k_\infty = (T/300.0)^{0.0}$ $f_c = (0.0)$
43	$\text{H}_3\text{O}^+(\text{H}_2\text{O})_3 + \text{H}_2\text{O} \longrightarrow$ $\text{H}_3\text{O}^+(\text{H}_2\text{O})_4 + \text{M}$	$k_0 = M(9.0\text{e-}28)(T/300.0)^{0.0}$ $k_\infty = (T/300.0)^{0.0}$ $f_c = (0.0)$
44	$\text{H}_3\text{O}^+(\text{H}_2\text{O})_3 + \text{M} \longrightarrow$ $\text{H}_3\text{O}^+(\text{H}_2\text{O})_2 + \text{H}_2\text{O} + \text{M}$	$k_0 = M(4.0\text{e-}14)(T/300.0)^{0.0}$ $k_\infty = (T/300.0)^{0.0}$ $f_c = (0.0)$
45	$\text{H}_3\text{O}^+(\text{H}_2\text{O})_4 + \text{M} \longrightarrow$ $\text{H}_3\text{O}^+(\text{H}_2\text{O})_3 + \text{H}_2\text{O} + \text{M}$	$k_0 = M(6.0\text{e-}12)(T/300.0)^{0.0}$ $k_\infty = (T/300.0)^{0.0}$ $f_c = (0.0)$

46	$\text{O}_2 + \text{e}^- \longrightarrow \text{O}_2^- + \text{M}$	$k_0 = M(2.0\text{e-}31) \exp(-600.0/T)(T/300.0)^{1.0}$ $k_\infty = (T/300.0)^{0.0}$ $f_c = (0.0)$
47	$\text{O}^- + \text{CO}_2 \longrightarrow$ $\text{CO}_3^- + \text{CO}_2 + \text{M}$	$k_0 = M(1.1\text{e-}27)(T/300.0)^{0.0}$ $k_\infty = (T/300.0)^{0.0}$ $f_c = (0.0)$
48	$\text{O}_2^- + \text{CO}_2 \longrightarrow \text{CO}_4^- + \text{M}$	$k_0 = M(1.3\text{e-}29)(T/300.0)^{0.0}$ $k_\infty = (T/300.0)^{0.0}$ $f_c = (0.0)$
49	$\text{O}_2^+ + \text{O}^- \longrightarrow$ $\text{O}_2 + \text{O}(^3\text{P}) + \text{M}$	$k_0 = M(3.0\text{e-}25)(T/300.0)^{-2.5}$ $k_\infty = (T/300.0)^{0.0}$ $f_c = (0.0)$
50	$\text{O}_2^+ + \text{O}_2^- \longrightarrow 2\text{O}_2 + \text{M}$	$k_0 = M(3.0\text{e-}25)(T/300.0)^{-2.5}$ $k_\infty = (T/300.0)^{0.0}$ $f_c = (0.0)$

51	$\text{O}_2^+ + \text{O}_3^- \longrightarrow$ $\text{O}_2 + \text{O}_3 + \text{M}$	$k_0 = M(3.0\text{e-}25)(T/300.0)^{-2.5}$ $k_\infty = (T/300.0)^{0.0}$ $f_c = (0.0)$
52	$\text{O}_2^+ + \text{NO}_2^- \longrightarrow$ $\text{O}_2 + \text{NO}_2 + \text{M}$	$k_0 = M(3.0\text{e-}25)(T/300.0)^{-2.5}$ $k_\infty = (T/300.0)^{0.0}$ $f_c = (0.0)$
53	$\text{O}_2^+ + \text{NO}_3^- \longrightarrow$ $\text{O}_2 + \text{NO}_3 + \text{M}$	$k_0 = M(3.0\text{e-}25)(T/300.0)^{-2.5}$ $k_\infty = (T/300.0)^{0.0}$ $f_c = (0.0)$
54	$\text{O}_2^+ + \text{CO}_3^- \longrightarrow$ $\text{O}_2 + \text{CO}_2 + \text{O}(^3\text{P}) + \text{M}$	$k_0 = M(3.0\text{e-}25)(T/300.0)^{-2.5}$ $k_\infty = (T/300.0)^{0.0}$ $f_c = (0.0)$
55	$\text{O}_2^+ + \text{NO}_2^-(\text{H}_2\text{O})_2 \longrightarrow$ $\text{O}_2 + \text{NO}_2 + 2\text{H}_2\text{O} + \text{M}$	$k_0 = M(1.0\text{e-}25)(T/300.0)^{-2.5}$ $k_\infty = (T/300.0)^{0.0}$ $f_c = (0.0)$

56	$\text{NO}^+ + \text{O}^- \longrightarrow$ $\text{NO} + \text{O}(^3\text{P}) + \text{M}$	$k_0 = M(1.0\text{e-}25)(T/300.0)^{-2.5}$ $k_\infty = (T/300.0)^{0.0}$ $f_c = (0.0)$
57	$\text{NO}^+ + \text{O}_2^- \longrightarrow$ $\text{NO} + \text{O}_2 + \text{M}$	$k_0 = M(1.0\text{e-}25)(T/300.0)^{-2.5}$ $k_\infty = (T/300.0)^{0.0}$ $f_c = (0.0)$
58	$\text{NO}^+ + \text{O}_3^- \longrightarrow$ $\text{NO} + \text{O}_3 + \text{M}$	$k_0 = M(1.0\text{e-}25)(T/300.0)^{-2.5}$ $k_\infty = (T/300.0)^{0.0}$ $f_c = (0.0)$
59	$\text{NO}^+ + \text{NO}_2^- \longrightarrow$ $\text{NO} + \text{NO}_2 + \text{M}$	$k_0 = M(1.0\text{e-}25)(T/300.0)^{-2.5}$ $k_\infty = (T/300.0)^{0.0}$ $f_c = (0.0)$
60	$\text{NO}^+ + \text{NO}_3^- \longrightarrow$ $\text{NO} + \text{NO}_3 + \text{M}$	$k_0 = M(1.0\text{e-}25)(T/300.0)^{-2.5}$ $k_\infty = (T/300.0)^{0.0}$ $f_c = (0.0)$

61	$\text{NO}^+ + \text{CO}_3^- \longrightarrow$ $\text{NO} + \text{CO}_2 + \text{O}(^3\text{P}) + \text{M}$	$k_0 = M(1.0\text{e-}25)(T/300.0)^{-2.5}$ $k_\infty = (T/300.0)^{0.0}$ $f_c = (0.0)$
62	$\text{NO}^+ + \text{NO}_2^-(\text{H}_2\text{O})_2 \longrightarrow$ $\text{NO} + \text{NO}_2 + 2\text{H}_2\text{O} + \text{M}$	$k_0 = M(1.0\text{e-}25)(T/300.0)^{-2.5}$ $k_\infty = (T/300.0)^{0.0}$ $f_c = (0.0)$
63	$\text{NO}^+ + \text{CO}_4^- \longrightarrow$ $\text{NO} + \text{CO}_2 + \text{O}_2 + \text{M}$	$k_0 = M(1.0\text{e-}25)(T/300.0)^{-2.5}$ $k_\infty = (T/300.0)^{0.0}$ $f_c = (0.0)$
64	$\text{CO}_2^+ + \text{O}_2^- \longrightarrow$ $\text{CO}_2 + \text{O}_2 + \text{M}$	$k_0 = M(1.0\text{e-}25)(T/300.0)^{-2.5}$ $k_\infty = (T/300.0)^{0.0}$ $f_c = (0.0)$
65	$\text{CO}_2^+ + \text{O}_3^- \longrightarrow$ $\text{CO}_2 + \text{O}_3 + \text{M}$	$k_0 = M(1.0\text{e-}25)(T/300.0)^{-2.5}$ $k_\infty = (T/300.0)^{0.0}$ $f_c = (0.0)$

66	$\text{CO}_2^+ + \text{NO}_2^- \longrightarrow$ $\text{CO}_2 + \text{NO}_2 + \text{M}$	$k_0 = M(1.0\text{e-}25)(T/300.0)^{-2.5}$ $k_\infty = (T/300.0)^{0.0}$ $f_c = (0.0)$
67	$\text{CO}_2^+ + \text{NO}_3^- \longrightarrow$ $\text{CO}_2 + \text{NO}_3 + \text{M}$	$k_0 = M(1.0\text{e-}25)(T/300.0)^{-2.5}$ $k_\infty = (T/300.0)^{0.0}$ $f_c = (0.0)$
68	$\text{CO}_2^+ + \text{CO}_3^- \longrightarrow$ $\text{CO}_2 + \text{CO}_2 + \text{O}({}^3\text{P}) + \text{M}$	$k_0 = M(1.0\text{e-}25)(T/300.0)^{-2.5}$ $k_\infty = (T/300.0)^{0.0}$ $f_c = (0.0)$
69	$\text{CO}_2^+ + \text{NO}_2^-(\text{H}_2\text{O})_2 \longrightarrow$ $\text{CO}_2 + \text{NO}_2 + 2\text{H}_2\text{O} + \text{M}$	$k_0 = M(1.0\text{e-}25)(T/300.0)^{-2.5}$ $k_\infty = (T/300.0)^{0.0}$ $f_c = (0.0)$
70	$\text{CO}_2^+ + \text{CO}_4^- \longrightarrow$ $2\text{CO}_2 + \text{O}_2 + \text{M}$	$k_0 = M(1.0\text{e-}25)(T/300.0)^{-2.5}$ $k_\infty = (T/300.0)^{0.0}$ $f_c = (0.0)$

71	$\text{H}_3\text{O}^+ + \text{O}^- \longrightarrow$ $\text{H}_2 + \text{O}(^3\text{P}) + \text{OH} + \text{M}$	$k_0 = M(3.0\text{e-}25)(T/300.0)^{-2.5}$ $k_\infty = (T/300.0)^{0.0}$ $f_c = (0.0)$
72	$\text{H}_3\text{O}^+ + \text{O}_2^- \longrightarrow$ $\text{H}_2 + \text{O}_2 + \text{OH} + \text{M}$	$k_0 = M(3.0\text{e-}25)(T/300.0)^{-2.5}$ $k_\infty = (T/300.0)^{0.0}$ $f_c = (0.0)$
73	$\text{H}_3\text{O}^+ + \text{O}_3^- \longrightarrow$ $\text{H}_2 + \text{O}_3 + \text{OH} + \text{M}$	$k_0 = M(3.0\text{e-}25)(T/300.0)^{-2.5}$ $k_\infty = (T/300.0)^{0.0}$ $f_c = (0.0)$
74	$\text{H}_3\text{O}^+ + \text{NO}_2^- \longrightarrow$ $\text{H}_2 + \text{NO}_2 + \text{OH} + \text{M}$	$k_0 = M(3.0\text{e-}25)(T/300.0)^{-2.5}$ $k_\infty = (T/300.0)^{0.0}$ $f_c = (0.0)$
75	$\text{H}_3\text{O}^+ + \text{NO}_3^- \longrightarrow$ $\text{H}_2 + \text{NO}_3 + \text{OH} + \text{M}$	$k_0 = M(3.0\text{e-}25)(T/300.0)^{-2.5}$ $k_\infty = (T/300.0)^{0.0}$ $f_c = (0.0)$

76	$\text{H}_3\text{O}^+ + \text{CO}_3^- \longrightarrow$ $\text{H}_2 + \text{CO}_2 + \text{O}(^3\text{P}) + \text{OH} + \text{M}$	$k_0 = M(3.0\text{e-}25)(T/300.0)^{-2.5}$ $k_\infty = (T/300.0)^{0.0}$ $f_c = (0.0)$
77	$\text{H}_3\text{O}^+ + \text{NO}_2^-(\text{H}_2\text{O})_2 \longrightarrow$ $\text{H}_2 + \text{NO}_2 + 2\text{H}_2\text{O} + \text{OH} + \text{M}$	$k_0 = M(1.0\text{e-}25)(T/300.0)^{-2.5}$ $k_\infty = (T/300.0)^{0.0}$ $f_c = (0.0)$
78	$\text{H}_3\text{O}^+ + \text{CO}_4^- \longrightarrow$ $\text{H}_2 + \text{CO}_2 + \text{O}_2 + \text{OH} + \text{M}$	$k_0 = M(3.0\text{e-}25)(T/300.0)^{-2.5}$ $k_\infty = (T/300.0)^{0.0}$ $f_c = (0.0)$
79	$\text{H}_3\text{O}^+(\text{H}_2\text{O}) + \text{O}^- \longrightarrow$ $\text{H}_2 + \text{O}(^3\text{P}) + \text{OH} + \text{H}_2\text{O} + \text{M}$	$k_0 = M(1.0\text{e-}25)(T/300.0)^{-2.5}$ $k_\infty = (T/300.0)^{0.0}$ $f_c = (0.0)$
80	$\text{H}_3\text{O}^+(\text{H}_2\text{O}) + \text{O}_2^- \longrightarrow$ $\text{H}_2 + \text{O}_2 + \text{OH} + \text{H}_2\text{O} + \text{M}$	$k_0 = M(1.0\text{e-}25)(T/300.0)^{-2.5}$ $k_\infty = (T/300.0)^{0.0}$ $f_c = (0.0)$

81	$\text{H}_3\text{O}^+(\text{H}_2\text{O}) + \text{O}_3^- \longrightarrow$ $\text{H}_2 + \text{O}_3 + \text{OH} + \text{H}_2\text{O} + \text{M}$	$k_0 = M(1.0\text{e-}25)(T/300.0)^{-2.5}$ $k_\infty = (T/300.0)^{0.0}$ $f_c = (0.0)$
82	$\text{H}_3\text{O}^+(\text{H}_2\text{O}) + \text{NO}_2^- \longrightarrow$ $\text{H}_2 + \text{NO}_2 + \text{OH} + \text{H}_2\text{O} + \text{M}$	$k_0 = M(1.0\text{e-}25)(T/300.0)^{-2.5}$ $k_\infty = (T/300.0)^{0.0}$ $f_c = (0.0)$
83	$\text{H}_3\text{O}^+(\text{H}_2\text{O}) + \text{NO}_3^- \longrightarrow$ $\text{H}_2 + \text{NO}_3 + \text{OH} + \text{H}_2\text{O} + \text{M}$	$k_0 = M(1.0\text{e-}25)(T/300.0)^{-2.5}$ $k_\infty = (T/300.0)^{0.0}$ $f_c = (0.0)$
84	$\text{H}_3\text{O}^+(\text{H}_2\text{O}) + \text{CO}_3^- \longrightarrow$ $\text{H}_2 + \text{CO}_2 + \text{O}(^3\text{P}) + \text{OH} + \text{H}_2\text{O} + \text{M}$	$k_0 = M(1.0\text{e-}25)(T/300.0)^{-2.5}$ $k_\infty = (T/300.0)^{0.0}$ $f_c = (0.0)$
85	$\text{H}_3\text{O}^+(\text{H}_2\text{O}) + \text{NO}_2^-(\text{H}_2\text{O})_2 \longrightarrow$ $\text{H}_2 + \text{NO}_2 + \text{OH} + 3\text{H}_2\text{O} + \text{M}$	$k_0 = M(1.0\text{e-}25)(T/300.0)^{-2.5}$ $k_\infty = (T/300.0)^{0.0}$ $f_c = (0.0)$

86	$\text{H}_3\text{O}^+(\text{H}_2\text{O}) + \text{CO}_4^- \longrightarrow$ $\text{H}_2 + \text{CO}_2 + \text{O}_2 + \text{OH} + \text{H}_2\text{O} + \text{M}$	$k_0 = M(1.0\text{e-}25)(T/300.0)^{-2.5}$ $k_\infty = (T/300.0)^{0.0}$ $f_c = (0.0)$
87	$\text{O}_2^+\text{O}_2 + \text{O}^- \longrightarrow$ $\text{O}_2 + \text{O}({}^3\text{P}) + \text{O}_2 + \text{M}$	$k_0 = M(1.0\text{e-}25)(T/300.0)^{-2.5}$ $k_\infty = (T/300.0)^{0.0}$ $f_c = (0.0)$
88	$\text{O}_2^+\text{O}_2 + \text{O}_2^- \longrightarrow 3\text{O}_2 + \text{M}$	$k_0 = M(1.0\text{e-}25)(T/300.0)^{-2.5}$ $k_\infty = (T/300.0)^{0.0}$ $f_c = (0.0)$
89	$\text{O}_2^+\text{O}_2 + \text{NO}_3^- \longrightarrow$ $2\text{O}_2 + \text{NO}_3 + \text{M}$	$k_0 = M(1.0\text{e-}25)(T/300.0)^{-2.5}$ $k_\infty = (T/300.0)^{0.0}$ $f_c = (0.0)$
90	$\text{O}_2^+\text{O}_2 + \text{CO}_3^- \longrightarrow$ $2\text{O}_2 + \text{CO}_2 + \text{O}({}^3\text{P}) + \text{M}$	$k_0 = M(1.0\text{e-}25)(T/300.0)^{-2.5}$ $k_\infty = (T/300.0)^{0.0}$ $f_c = (0.0)$

91	$\text{O}_2^+ \text{O}_2 + \text{NO}_2^- (\text{H}_2\text{O})_2 \longrightarrow$ $2\text{O}_2 + \text{NO}_2 + 2\text{H}_2\text{O} + \text{M}$	$k_0 = M(1.0\text{e-}25)(T/300.0)^{-2.5}$ $k_\infty = (T/300.0)^{0.0}$ $f_c = (0.0)$
92	$\text{O}_2^+ \text{O}_2 + \text{CO}_4^- \longrightarrow$ $3\text{O}_2 + \text{CO}_2 + \text{M}$	$k_0 = M(1.0\text{e-}25)(T/300.0)^{-2.5}$ $k_\infty = (T/300.0)^{0.0}$ $f_c = (0.0)$
93	$\text{O}_2^- + \text{O}_2 \longrightarrow \text{O}_4^- + \text{M}$	$k_0 = M(3.4\text{e-}31)(T/300.0)^{0.0}$ $k_\infty = (T/300.0)^{0.0}$ $f_c = (0.0)$
94	$\text{O}_2^- + \text{H}_2\text{O} \longrightarrow \text{O}_2^- (\text{H}_2\text{O}) + \text{M}$	$k_0 = M(2.2\text{e-}28)(T/300.0)^{0.0}$ $k_\infty = (T/300.0)^{0.0}$ $f_c = (0.0)$
95	$\text{O}_2^- (\text{H}_2\text{O}) + \text{NO} \longrightarrow$ $\text{NO}_3^- + \text{H}_2\text{O} + \text{M}$	$k_0 = M(T/300.0)^{0.0}$ $k_\infty = (2.0\text{e-}10)(T/300.0)^{0.0}$ $f_c = (0.0)$

A.1.3 Photolysis Reactions

#	Photolysis Reaction
1	$\text{Br}_2 + h\nu \longrightarrow \text{Br} + \text{Br}$
2	$\text{BrCl} + h\nu \longrightarrow \text{Cl} + \text{Br}$
3	$\text{BrO} + h\nu \longrightarrow \text{O}({}^3\text{P}) + \text{Br}$
4	$\text{BrO} + h\nu \longrightarrow \text{Br} + \text{O}({}^1\text{D})$
5	$\text{BrONO} + h\nu \longrightarrow \text{Br} + \text{NO}_2$
6	$\text{BrONO}_2 + h\nu \longrightarrow \text{NO}_2 + \text{BrO}$
7	$\text{BrONO}_2 + h\nu \longrightarrow \text{Br} + \text{NO}_3$

8	$\text{CF}_2\text{Cl}_2 + h\nu \longrightarrow \text{Cl} + \text{Cl}$
9	$\text{CH}_3\text{Br} + h\nu \longrightarrow \text{Br} + \text{CH}_3$
10	$\text{CH}_4 + h\nu \longrightarrow \text{H} + \text{CH}_3$
11	$\text{CO}_2 + h\nu \longrightarrow \text{O}({}^3\text{P}) + \text{CO}$
12	$\text{CO}_2 + h\nu \longrightarrow \text{CO} + \text{O}({}^1\text{D})$
13	$\text{Cl}_2 + h\nu \longrightarrow \text{Cl} + \text{Cl}$
14	$\text{Cl}_2\text{O}_2 + h\nu \longrightarrow \text{Cl} + \text{ClOO}$

15	$\text{ClNO}_2 + h\nu \longrightarrow \text{Cl} + \text{NO}_2$
16	$\text{ClO} + h\nu \longrightarrow \text{O}({}^3\text{P}) + \text{Cl}$
17	$\text{ClONO}_2 + h\nu \longrightarrow \text{Cl} + \text{NO}_3$
18	$\text{ClOO} + h\nu \longrightarrow \text{O}({}^3\text{P}) + \text{ClO}$
19	$\text{H}_2\text{O} + h\nu \longrightarrow$ $\text{H} + \text{H} + \text{O}({}^3\text{P})$
20	$\text{H}_2\text{O} + h\nu \longrightarrow \text{H}_2 + \text{O}({}^1\text{D})$
21	$\text{H}_2\text{O} + h\nu \longrightarrow \text{H} + \text{OH}$

22	$\text{H}_2\text{O} + h\nu \longrightarrow \text{H}_2 + \text{O}^+$
23	$\text{H}_2\text{O}_2 + h\nu \longrightarrow \text{OH} + \text{OH}$
24	$\text{HCHO} + h\nu \longrightarrow \text{H}_2 + \text{CO}$
25	$\text{HCHO} + h\nu \longrightarrow \text{H} + \text{HCO}$
26	$\begin{aligned} \text{HCHO} + h\nu &\longrightarrow \\ \text{H} + \text{H} + \text{CO} \end{aligned}$
27	$\text{HCl} + h\nu \longrightarrow \text{H} + \text{Cl}$
28	$\text{HO}_2 + h\nu \longrightarrow \text{O}({}^3\text{P}) + \text{OH}$

29	$\text{HO}_2\text{NO}_2 + h\nu \longrightarrow \text{NO}_2 + \text{HO}_2$
30	$\text{HO}_2\text{NO}_2 + h\nu \longrightarrow \text{OH} + \text{NO}_3$
31	$\text{HOBr} + h\nu \longrightarrow \text{OH} + \text{Br}$
32	$\text{HOCl} + h\nu \longrightarrow \text{Cl} + \text{OH}$
33	$\text{HONO} + h\nu \longrightarrow \text{NO} + \text{OH}$
34	$\text{HNO}_3 + h\nu \longrightarrow \text{OH} + \text{NO}_2$
35	$\text{HNO}_3 + h\nu \longrightarrow \text{O}(^3\text{P}) + \text{HONO}$

36	$\text{CH}_3\text{O}_2\text{NO}_2 + h\nu \longrightarrow \text{NO}_2 + \text{CH}_3\text{O}_2$
37	$\text{CH}_3\text{O}_2\text{NO}_2 + h\nu \longrightarrow \text{NO}_3 + \text{CH}_3\text{O}$
38	$\text{CH}_3\text{OCl} + h\nu \longrightarrow \text{Cl} + \text{CH}_3\text{O}$
39	$\text{CH}_3\text{OH} + h\nu \longrightarrow \text{H}_2 + \text{HCHO}$
40	$\text{CH}_3\text{OH} + h\nu \longrightarrow \text{OH} + \text{CH}_3$
41	$\text{CH}_3\text{ONO}_2 + h\nu \longrightarrow \text{NO}_2 + \text{CH}_3\text{O}$
42	$\text{CH}_3\text{OOH} + h\nu \longrightarrow \text{OH} + \text{CH}_3\text{O}$

43	$\text{CH}_3\text{OOH} + h\nu \longrightarrow$ $\text{O}(^3\text{P}) + \text{H} + \text{CH}_3\text{O}$
44	$\text{N}_2\text{O} + h\nu \longrightarrow \text{N}_2 + \text{O}(^1\text{D})$
45	$\text{N}_2\text{O}_5 + h\nu \longrightarrow$ $\text{O}(^3\text{P}) + \text{NO} + \text{NO}_3$
46	$\text{N}_2\text{O}_5 + h\nu \longrightarrow \text{NO}_2 + \text{NO}_3$
47	$\text{NO} + h\nu \longrightarrow \text{O}(^3\text{P}) + \text{N}$
48	$\text{NO}_2 + h\nu \longrightarrow \text{NO} + \text{O}(^1\text{D})$
49	$\text{NO}_2 + h\nu \longrightarrow \text{O}(^3\text{P}) + \text{NO}$

50	$\text{NO}_3 + h\nu \longrightarrow \text{O}({}^3\text{P}) + \text{NO}_2$
51	$\text{NO}_3 + h\nu \longrightarrow \text{O}_2 + \text{NO}$
52	$\text{O}_2 + h\nu \longrightarrow \text{O}({}^3\text{P}) + \text{O}({}^1\text{D})$
53	$\text{O}_2 + h\nu \longrightarrow \text{O}({}^3\text{P}) + \text{O}({}^3\text{P})$
54	$\text{O}_3 + h\nu \longrightarrow \text{O}({}^3\text{P}) + \text{O}_2$
55	$\text{O}_3 + h\nu \longrightarrow \text{O}_2 + \text{O}({}^1\text{D})$
56	$\text{OCIO} + h\nu \longrightarrow \text{O}({}^3\text{P}) + \text{ClO}$

57	$\text{OCIO} + h\nu \longrightarrow \text{ClO} + \text{O}({}^1\text{D})$
----	--

A.2 Master Chemical Mechanism

REFERENCES

- Ahmed, S. E., S. Pawar, and O. San (2020). Pyda: A hands-on introduction to dynamical data assimilation with python. *Fluids* 5(4), 225.
- Altmeyer, P. (2023). Conformal prediction.
- Arnaut, L. and H. Burrows (2006). *Chemical kinetics: from molecular structure to chemical reactivity*. Elsevier.
- Bishop, C. M., M. Svensén, and C. K. Williams (1998). Gtm: The generative topographic mapping. *Neural computation* 10(1), 215–234.
- Boldi, R. A. (1994). A model of the ion chemistry of electrified convection.
- Brook, R. D., S. Rajagopalan, C. A. Pope III, J. R. Brook, A. Bhatnagar, A. V. Diez-Roux, F. Holguin, Y. Hong, R. V. Luepker, M. A. Mittleman, et al. (2010). Particulate matter air pollution and cardiovascular disease. *Circulation* 121(21), 2331–2378.
- Brunton, S. L., B. W. Brunton, J. L. Proctor, E. Kaiser, and J. N. Kutz (2017). Chaos as an intermittently forced linear system. *Nature communications* 8(1), 19.
- Burkholder, J., S. Sander, J. Abbatt, J. Barker, C. Cappa, J. Crounse, T. Dibble, R. Huie, C. Kolb, M. Kurylo, et al. (2020). Chemical kinetics and photochemical data for use in atmospheric studies; evaluation number 19. Technical report, Pasadena, CA: Jet Propulsion Laboratory, National Aeronautics and Space ...
- Buyantuyev, A. and J. Wu (2017). Remote sensing applications for land cover and land-use transformations in semiarid and arid environments. *Journal of Arid Environments* 140, 1–5.
- Carleo, G., K. Choo, J. Hofmann, E. Huang, C. Hughes, M. Hush, R. Iten, J. McClean, C. Miles, J. Preskill, et al. (2019). Machine learning and the physical sciences. *Reviews of Modern Physics* 91(4), 045002.
- Centre for Ecology and Hydrology (2017). Ecological sensing: a revolution in biodiversity monitoring.
- Chen, J., X. Shi, X. Li, M. Wang, W. Shen, and Y. Liu (2019). A review of air quality modeling: From gas-phase to particulate matter. *Advances in Atmospheric Sciences* 36(10), 921–947.
- Clark, M. P., W. N. Adger, S. Dessai, M. Goulden, D. W. Cash, and R. a. N. M. a. A. Stern, Nicholas and Gonzalez (2016). Urbanization, climate change and economic growth: Challenges and opportunities for policy makers. *Science of the Total Environment* 557–558, 279–291.

- Costello, A., M. Abbas, A. Allen, S. Ball, S. Bell, R. Bellamy, S. Friel, N. Groce, A. Johnson, M. Kett, M. Lee, C. Levy, M. Maslin, D. McCoy, B. McGuire, H. Montgomery, D. Napier, C. Pagel, J. Patel, J. A. P. de Oliveira, N. Redclift, H. Rees, D. Rogger, J. Scott, J. Stephenson, J. Twigg, J. Wolff, and C. Patterson (2009). Managing the health effects of climate change: Lancet and university college london institute for global health commission. *The Lancet* 373(9676), 1693–1733.
- DeLucia, E. H., N. Gomez-Casanovas, S. P. Long, M. A. Mayes, R. A. Montgomery, W. J. Parton, W. J. Sacks, J. P. Schimel, J. Verfaillie, and W. L. Silver (2021). The missing soil n: detecting processes driving soil nitrogen storage in complex ecosystems. *Journal of Ecology* 109(2), 447–459.
- Edenhofer, O., R. Pichs-Madruga, Y. Sokona, E. Farahani, S. Kadner, K. Seyboth, A. Adler, I. Baum, S. Brunner, P. Eickemeier, et al. (2014). *Climate change 2014: Mitigation of climate change*. Cambridge University Press.
- Field, C., V. Barros, D. Dokken, K. Mach, M. Mastrandrea, T. Bilir, M. Chatterjee, K. Ebi, Y. Estrada, R. Genova, et al. (2014). *Climate change 2014: Impacts, adaptation, and vulnerability. Part A: Global and sectoral aspects*. Cambridge University Press.
- Friedlingstein, P., M. W. Jones, M. O’Sullivan, R. M. Andrew, J. Hauck, G. P. Peters, W. Peters, J. Pongratz, S. Sitch, C. Le Quéré, et al. (2020). Global carbon budget 2020. *Earth System Science Data* 12(4), 3269–3340.
- Gamon, J. A., K. F. Huemmrich, R. S. Stone, and C. E. Tweedie (2016). Spatial and temporal variation in primary productivity (ndvi) of coastal alaskan tundra: Decreased vegetation growth following earlier snowmelt. *Remote Sensing of Environment* 175, 233–242.
- Haines, A., R. S. Kovats, D. Campbell-Lendrum, and C. Corvalán (2006). Climate change and human health: Impacts, vulnerability and public health. *Public Health* 120(7), 585–596.
- Hantson, S., A. Arneth, S. P. Harrison, D. I. Kelley, I. C. Prentice, S. S. Rabin, S. Archibald, F. Mouillot, S. R. Arnold, P. Artaxo, et al. (2016). The status and challenge of global fire modelling. *Biogeosciences* 13(11), 3359–3375.
- Houghton, J., Y. Ding, D. Griggs, M. Noguer, P. van der Linden, X. Dai, K. Maskell, and C. Johnson (2001). *Climate change 2001: The scientific basis*. Cambridge University Press.
- Houghton, J., G. Jenkins, and J. Ephraums (1990). *Climate change: The IPCC scientific assessment*. Cambridge University Press.
- Houghton, J., L. Meira Filho, B. Callander, N. Harris, A. Kattenberg, and K. Maskell (1996). *Climate change 1995: The science of climate change*. Cambridge University Press.

- Huang, J., L. Yu, J. Guo, X. Guo, W. Wang, C. Liu, and D. Ji (2017). Assessment of global surface energy budget datasets using flux tower observations. *Journal of Geophysical Research: Atmospheres* 122(14), 7452–7475.
- Innes, M., A. Edelman, K. Fischer, C. Rackauckas, E. Saba, V. B. Shah, and W. Tebbutt (2019). A differentiable programming system to bridge machine learning and scientific computing. *arXiv preprint arXiv:1907.07587*.
- Kelly, F. J. and J. C. Fussell (2011). Air pollution and public health: emerging hazards and improved understanding of risk. *Environmental Geochemistry and Health* 33(4), 363–373.
- Kohonen, T. (1982). Self-organized formation of topologically correct feature maps. *Biological cybernetics* 43(1), 59–69.
- Masson-Delmotte, V., P. Zhai, H.-O. Pörtner, D. Roberts, J. Skea, P. Shukla, A. Pirani, W. Moufouma-Okia, C. Péan, R. Pidcock, et al. (2018). *Global warming of 1.5°C. An IPCC special report on the impacts of global warming of 1.5°C above pre-industrial levels and related global greenhouse gas emission pathways, in the context of strengthening the global response to the threat of climate change, sustainable development, and efforts to eradicate poverty*. Intergovernmental Panel on Climate Change.
- Metz, B., O. Davidson, P. Bosch, R. Dave, and L. Meyer (2007). *Climate change 2007: Mitigation of climate change*. Cambridge University Press.
- National Research Council (2010). Verifying greenhouse gas emissions: Methods to support international climate agreements.
- Neumark, D. M. (1992). Transition state spectroscopy of bimolecular chemical reactions. *Annual Review of Physical Chemistry* 43(1), 153–176.
- Oleson, K., D. Lawrence, G. Bonan, and M. Flanner (2013). Interactions between land use change and carbon cycle feedbacks. *Global Biogeochemical Cycles* 27(4), 972–983.
- Parry, M., O. Canziani, J. Palutikof, P. van der Linden, and C. Hanson (2007). *Climate change 2007: Impacts, adaptation and vulnerability*. Cambridge University Press.
- Partridge, H., C. W. Bauschlicher Jr, J. R. Stallcop, and E. Levin (1993). Ab initio potential energy surface for h–h₂. *The Journal of chemical physics* 99(8), 5951–5960.
- Pasher, J., B.-J. Park, J. Théau, F. Pimont, and S. Goetz (2019). Remote sensing of wetlands: An overview and practical guide. *Wetlands Ecology and Management* 27(2), 129–147.
- Rackauckas, C., D. Kelly, Q. Nie, J. Li, C. Warner, M. Dhairya, J. Fang, E. Zhou, R. Supekar, S. Sandhu, et al. (2020). Universal differential equations for scientific machine learning. *arXiv preprint arXiv:2012.09345*.

- Raissi, M. and G. E. Karniadakis (2021). Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational Physics* 378, 686–707.
- Raissi, M., P. Perdikaris, and G. E. Karniadakis (2019). Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational Physics* 378, 686–707.
- Rasmussen, C. E., C. K. Williams, et al. (2006). *Gaussian processes for machine learning*, Volume 1. Springer.
- Solomon, S., D. Qin, M. Manning, Z. Chen, M. Marquis, K. Averyt, M. Tignor, and H. Miller Jr (2007). *Climate change 2007: The physical science basis*. Cambridge University Press.
- Stocker, T., D. Qin, G.-K. Plattner, M. Tignor, S. Allen, J. Boschung, A. Nauels, Y. Xia, V. Bex, and P. Midgley (2013). *Climate change 2013: The physical science basis*. Cambridge University Press.
- Thenkabail, P. S. (2019). Remote sensing of global croplands for food security. *Remote Sensing* 11(10), 1261.
- United Nations (2015). Transforming our world: the 2030 agenda for sustainable development. *UN General Assembly*.
- United Nations Environment Programme (2017). Adaptation gap report 2017.
- Wang, D., D. Xie, Y. Xie, and C. Chen (2017). Remote sensing applications for urban water resources: A review. *Remote Sensing* 9(8), 829.
- World Health Organization (2018). Climate change and health.
- Wu, J. and X. Zhang (2021). A review on physics-informed machine learning: Basic principles, recent developments and future directions. *Physics Reports* 903, 1–45.
- Wu, T., H.-J. Werner, and U. Manthe (2006). Accurate potential energy surface and quantum reaction rate calculations for the $\text{H} + \text{CH}_4 \rightarrow \text{H}_2 + \text{CH}_3$ reaction. *The Journal of chemical physics* 124(16).
- Xu, X., F. Deng, X. Guo, P. Lv, H. Zhong, Y. Hao, G. Hu, J. Huang, Y. Guo, Y. Liu, et al. (2017). Association between particulate matter air pollution and hospital admissions in patients with chronic obstructive pulmonary disease in Beijing, China. *Science of the Total Environment* 579, 1616–1621.

BIOGRAPHICAL SKETCH

UPDATE REQUIRED!!!

CURRICULUM VITAE

UPDATE REQUIRED!