THE UNIVERSITY OF QUEENSLAND

AUSTRALIA

Optimizing performance in Gaussian Processes

Michael Ciccotosto-Camp

Supervisor: Fred (Farbod) Roosta
Co-Supervisors: Andries Potgieter
Yan Zhao

Bachelor of Mathematics (Honours)
June 2022

The University of Queensland
School of Mathematics and Physics

Contents

Acknowledgements

I would like to deeply thank my supervisor Dr. Masoud Kamgarpour for his advice and all of his time spent with me. I consider myself lucky and am glad to have been his student for my honours year. I would also like to thank my co-supervisor Dr. Anna Puskás for the same reasons. A special thanks to Dr. Valentin Buciumas for his time spent teaching me while he was at The University of Queensland.

Introduction

Origins of group theory. The mathematical field of group theory has its origins in the early 19th century. At the time, mathematicians were investigating the solutions to polynomial equations. That is, solutions to equations of the form

$$a_n x^n + a_{n-1} x^{n-1} + \ldots + a_0 = 0.$$

Full solutions to polynomial equations of low degrees (i.e. $n \leq 4$) had already been formulated [Rig96]. These include the familiar *quadratic formula*, which has been known since antiquity. The formula tells us that the solutions to a general quadratic equation $ax^2 + bx + c = 0$ are given by

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}.$$

The full solutions to any cubic ($n = 3$) or quartic ($n = 4$) polynomial equation were also known. These are given by the lesser-known Cardano's formula and Ferrari's method, respectively. We say that a polynomial is *solvable by radicals* if one can write all of its solutions in terms of its coefficients combined with the algebraic operations; addition, subtraction, multiplication, division, powers and radicals (i.e. k^{th} roots).

In the 1830s, the mathematician Évariste Galois provided an elegant method to prove that a general polynomial of degree $n \geq 5$ is not solvable by radicals. Galois understood that to every polynomial one could associate a *Galois group*, a new mathematical object at the time. The Galois group was the first object in a class of mathematical objects that we call *groups* today. We say that the pair $(G, \circ)$ is a group, where G is a set and $\circ \colon G \times G \to G$ is a binary operation on G, when three conditions are satisfied:

- Associativity: $g \circ (h \circ k) = (g \circ h) \circ k$ for all $g, h, k \in G$.
- Existence of an identity: there exists some $1_G \in G$ such that $1_G \circ g = g \circ 1_G = g$ for all $g \in G$.
- Existence of inverses: for every $g \in G$, there exists some $g^{-1} \in G$ such that $g \circ g^{-1} = g^{-1} \circ g = 1_G$.

Examples of groups that are likely familiar to the reader include $(\mathbb{Z}, +)$, the integers under addition, $(\mathbb{R}^+, \times)$, the positive real numbers under multiplication, and $(\mathbb{Z}/n\mathbb{Z}, +)$, the integers modulo n under addition. Some geometric examples of groups are the *dihedral groups*. These groups are generated by the m symmetries associated to the regular m-sided polygon (i.e. a polygon with all interior angles and all side lengths the same). Then each dihedral group contains $2m$ elements (m reflections and m rotations) with the group operation of composition of reflections and rotations.

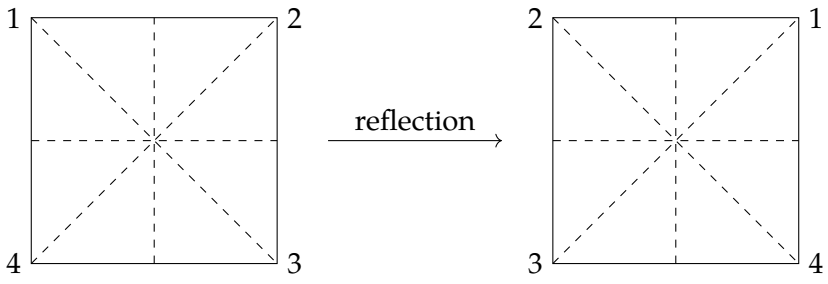


Figure 1. The symmetries of a square and a reflection about the vertical line of symmetry.

This gives us an intuitive understanding of groups: they encode the symmetries of mathematical objects.

What is representation theory? The study of groups yields insight into geometric objects. The action of the dihedral group on the m-gon serves as example of a group acting on a geometric object. More generally, we can consider the action of a group on some object. Specifically, we say that a group G *acts* on a set X if, for each $g \in G$, there is a map $\cdot\colon G \times X \to X$ satisfying $1_G \cdot x = x$ and $g \cdot (h \cdot x) = (gh) \cdot x$ for all $x \in X$. Alternatively, one can view this as a group homomorphism $\rho\colon G \to \mathrm{Sym}(X)$, where $\mathrm{Sym}(X)$ is the symmetric group associated to X, i.e. the group of permutations of elements of X.

Now we linearise the setting above by requiring that $X = V$ is a *vector space*. Then we say that G acts *linearly* on V if there exists a group homomorphism $\rho\colon G \to \mathrm{GL}(V)$. We call (V, ρ) a *representation* of G, and ρ is often suppresed from notation. We see that G acts on V in the sense that $\rho(g)\colon V \to V$ is a linear invertible map on V. We may denote $\rho(g)(v)$ by $g \cdot v$ as before.

Representation theory is concerned with understanding and classifying linear actions of groups. The general situation of representation theory is as follows. If the group G acts on a vector space V, then we say that a vector subspace $W \subseteq V$ is a *subrepresentation* of V if it is invariant under the action of G. A representation is called *irreducible* if its only proper subrepresentation is the trivial representation $W = \{0\}$. The primary goals of representation theory are finding all irreducible representations of G, and to decompose a given representation into its irreducible components.

We can think of irreducible representations as the building blocks of all other representations. This is a common idea in mathematics, seen in other areas. For instance, in number theory, the building blocks of integers are primes and, in group theory, the building blocks of groups are simple groups.

Writing a general representation in terms of irreducible components is not always possible. We call a representation *decomposable* if we can write it as the direct sum of irreducible representations. A lot can be said about the case where the representation of a finite group is over a field whose characteristic not dividing the order of the group. In this case, *Maschke's theorem* tells us that these representations are always decomposable [Lan02]. In particular, complex representations of a finite group are always decomposable.

Gelfand Pairs. Henceforth, we assume some knowledge of abstract algebra from the reader. Let G be a finite group and $K \leq G$ a subgroup. The pair (G, K) is called a *Gelfand pair* if the induced representation $\mathrm{Ind}_K^G \mathbf{1}$ is multiplicity-free. Here $\mathbf{1}$ denotes the trivial (1-dimensional) complex representation of K, and multiplicity-free means that any irreducible representation appears in the decomposition of $\mathrm{Ind}_K^G \mathbf{1}$ at most once (up to isomorphism).

Gelfand pairs play an important role in representation theory [Mus93], analysis [Kor80, Mor18], combinatorics [BI84], number theory [Gro91, Ter99] and probability [CSST20, Dia88]. One of our objectives is to give a detailed study of Gelfand pairs of finite groups. A main theorem of this thesis is the following:

Theorem 1. (*Gelfand's Trick*) *Let G be a finite group and K a subgroup of G. Suppose $\varphi\colon G \to G$ is an involutive anti-automorphism (i.e. a bijective anti-homomorphism) such that $K\varphi(x)K = KxK$ for all $x \in G$. Then (G, K) is a Gelfand pair.*

The theorem above is proved using the *Hecke algebra*. There are multiple constructions of Hecke algebras in the literature [CMHL03, CSST20].

Types of Hecke algebras. Another objective of this thesis is to present these a priori different Hecke algebras and resolve their apparent discrepancies. For instance, one way to define the Hecke algebra is as a convolution algebra of K-bi-invariant complex-valued functions $f : G \to \mathbb{C}$ on a group. Another way to define the Hecke algebra is as the algebra generated by $n-1$ variables $T_1, \dots T_{n-1}$ subject to a *quadratic relation* $T_i^2 = (q-1)T_i + q$ and a *braid relation*

$$\underbrace{T_iT_jT_i\dots}_{m_{ij} \text{ terms}} = \underbrace{T_jT_iT_j\dots}_{m_{ij} \text{ terms}}.$$

Here m_{ij} is the ij^{th} entry in the *Coxeter matrix* associated to the *Weyl group* of G. The name 'braid relation' is due to a method of visualising the *symmetric group* S_n. If n is a positive integer then the group S_n is the collection of bijections on the set $\{1, 2, \dots, n\}$ to itself, with the group operation of composing functions. A natural method of visualising elements and multiplication in this group is via *braid diagrams*. For instance, if $\sigma = (1\ 2)(3\ 5\ 4)$ and $\pi = (1\ 2\ 4\ 6\ 5\ 3)$ are permutations in S_6 (written in cycle notation), then we may visualise these elements and their product $\pi\sigma = (1\ 4)(5\ 6)$ in the following manner:

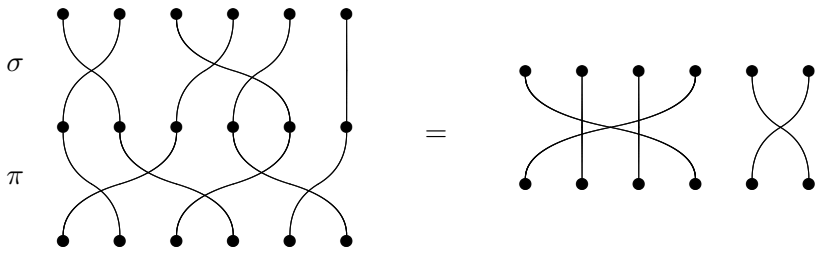


FIGURE 2. A braid diagram visualising the multiplication $\pi\sigma = (1\ 4)(5\ 6)$.

Why study Hecke algebras? The Hecke algebra arises naturally when one wishes to compute certain irreducible representations of a group [RW21, CMHL03]. Consider a finite group G and a normal subgroup $N \triangleleft G$. If G acts linearly on a vector space V (i.e. V is a representation of G), then there is a natural action of G on the subrepresentation V^N, the space of vectors in V that are fixed by N. Under this action, N will clearly act trivially on V^N. This yields a representation of the quotient group G/N. After some representation theoretic arguments, one arrives at the conclusion that

$$\left\{ \begin{array}{c} \text{Irreducible representations} \\ \text{of } G \text{ with } N\text{-fixed vectors} \end{array} \right\} \overset{1:1}{\longleftrightarrow} \left\{ \begin{array}{c} \text{Irreducible representations} \\ \text{of } G/N \end{array} \right\}.$$

It is a straightforward exercise that a complex representation of the group G/N is the same as a representation of the algebra $\mathbb{C}[G/N]$, the *group algebra* of G/N.

What happens when we do not require a normal subgroup of G? Consider an arbitrary subgroup K of a finite group G. Now G/K is no longer necessarily a group, so G/K and $\mathbb{C}[G/K]$ no longer necessarily make sense. We ask ourselves: what acts on V^K? The action of G on V is not well-defined on V^K since K is no longer normal. It is not obvious how we could study irreducible G-representations with K-fixed vectors. We are able to salvage the situation with the help of the Hecke algebra.

For $g \in G$, define the Hecke operator $[KgK] := \frac{1}{|K|}\sum_{x\in KgK} x \in \mathbb{C}[G]$, which acts on V^K by

$$[KgK]\cdot v := \frac{1}{|K|}\sum_{x\in KgK} x\cdot v.$$

Now define the Hecke algebra $\mathcal{H}(G,K)$ to be the space of functions $f\colon G\to\mathbb{C}$ that are constant on K-double cosets. The indicator functions χ_{KgK} form a basis of this space and we can uniquely associate the indicator functions χ_{KgK} to the Hecke operators $[KgK]$. We see that, through the Hecke operators, we have defined an action of $\mathcal{H}(G,K)$ on V^K. This answers our question of what acts on V^K. Through another representation-theoretic exercise, one can conclude that

$$\left\{\begin{array}{c}\text{Irreducible representations}\\ \text{of } G \text{ with } K\text{-fixed vectors}\end{array}\right\} \overset{1:1}{\longleftrightarrow} \left\{\begin{array}{c}\text{Irreducible representations}\\ \text{of } \mathcal{H}(G,K)\end{array}\right\}.$$

An immediate example of the utility of this result is as follows. It is easy to show that if $\mathcal{H}(G,K)$ is commutative, then all of its irreducible finite-dimensional representations are one-dimensional [EGH+11]. The commutativity of the Hecke algebra turns out to be an important property which will be investigated throughout this thesis.

Contents of this thesis. In Chapter 1, we begin our study of the Hecke algebra. First, we investigate the convolution algebra of all complex-valued functions on G and its ideal of K-right-invariant complex-valued functions. This is followed by results describing the relationship between the induced representation and its associated Hecke algebra. We use these results to prove Theorem 1. This allows us to write down simple proofs that $\mathrm{Ind}_K^G \mathbf{1}$ is multiplicity-free for certain choices of G and K. Namely, (G,K) with G commutative, (G,K) with $[G\colon K]=2$, $(S_{n+m}, S_n\times S_m)$, and $(O_{n+1}(\mathbb{F}_q), O_n(\mathbb{F}_q))$ for q odd.

In Chapter 3, we generalise the discussion of Chapter 1 to the case of a non-trivial *character* $\sigma\colon K\to\mathbb{C}^\times$. Here our goal is to obtain a twisted analogue of Theorem 1. To this end, we describe the basis of the Hecke algebra using the idea of *relevant orbits*. We state and prove the generalisation of Theorem 1. We apply the new theorem to a particular representation, the *Gelfand–Graev representation* of $\mathrm{GL}_n(\mathbb{F}_q)$, to show that it is multiplicity-free.

In Chapter **??**, we investigate the Hecke algebra of Chapter 1 under the particular choice of $G=\mathrm{SL}_n(\mathbb{F}_q)$ and $K=B(\mathbb{F}_q)$, the *Borel subgroup* of G, i.e. the subgroup of upper-triangular matrices. The *Weyl group* associated to G is introduced and shown to be isomorphic to S_n. Next, we perform some elementary matrix calculations which yields the surprising result above: the Hecke algebra may be written in terms of $n-1$ generators subject to the quadratic relation and the braid relations associated to W. This leads to a concluding discussion of Hecke algebras generated by any finite *Coxeter group*.

In Chapter 2, we generalise the results of earlier chapters to the case where G is no longer finite, but instead a locally compact topological group. This allows for an extension of the theory we have developed to more general groups and their Hecke algebras. To do this, we discuss how one can impose a topological structure on a group and supply examples to give some intuition for these types of groups. To define Hecke algebras of these groups, we require some measure theory. In particular, the convolution product on the Hecke algebra is defined in terms of an integral with respect to the *Haar measure*. We spend some time developing the theory of Haar measures for this purpose. We conclude with a discussion of how to

recover the Hecke algebra of a finite group from this new definition. In this chapter, we shall denote the Hecke algebra by $C_c(K\backslash G/K)$ to emphasise the non-finiteness of G.

In Chapter **??**, we take a look at some specific Hecke algebras of locally compact topological groups. In particular, we restrict our attention to the general linear group over a non-archimedian local field k and its ring of integers $\mathcal{O}$. We look at the *Spherical Hecke algebra*, formed when one considers $G = \mathrm{GL}_n(k)$ and $K = K^\circ := \mathrm{GL}_n(\mathcal{O})$, and the *Iwahori–Hecke algebra*, formed when one considers $G = \mathrm{GL}_n(\mathcal{O})$ and $K = I$, the *Iwahori subgroup*. In order to investigate these algebras, we must develop an understanding of these fields. We detail their definition, classification and structure.

The contents of this thesis may be visualised with the following diagram.

$$\begin{array}{ccccccc}
\overset{\text{Ch. 3}}{\mathcal{H}(G,K,\sigma)} & \xrightarrow{\sigma=\mathbf{1}} & \overset{\text{Ch. 1}}{\mathcal{H}(G,K)} & \xleftarrow{G \text{ finite}} & \overset{\text{Ch. 2}}{C_c(K\backslash G/K)} & \xrightarrow{K=I} & \overset{\text{Ch. ??}}{C_c(I\backslash G/I)} \\
 & & \Big\downarrow{\scriptstyle G \text{ Lie type over } \mathbb{F}_q} & & \Big\downarrow{\scriptstyle K=K^\circ} & & \\
 & & \underset{\text{Ch. ??}}{\mathcal{H}_q(W,S)} & & \underset{\text{Ch. ??}}{C_c(K^\circ\backslash G/K^\circ)} & &
\end{array}$$

FIGURE 3. The relationship diagram of this thesis.

Directions for future research. We assume the reader is familiar with the contents of this thesis. The modern study of Hecke algebras is largely focused on the *Iwahori–Hecke algebra*, which is also known as the *affine Hecke algebra*. This algebra is central to the study of representations of *reductive groups* over non-archimedian local fields (e.g. groups such as GL_n, SL_n, Sp_{2n} over fields such as $\mathbb{Q}_p$ or $\mathbb{F}_q(\!(t)\!)$).

Some topics relevant to the Iwahori–Hecke algebra include *Bernstein's presentation*, the *Iwahori–Matsumoto presentation* and the *Satake isomorphism* [HKP09]. Properties of the Iwahori–Hecke algebra such as these presentations may be viewed as a consequence of the *universal unramified principal series module*, which we now describe.

Fix a "nice" (i.e. split and connected) reductive group G (e.g. SL_n) over a non-archimedian local field k with ring of integers $\mathcal{O}$. Then write A to mean a *split maximal torus* of G and write N to mean the *unipotent radical* of a Borel subgroup of G that contains A. Also recall I is the Iwahori subgroup of G given in Chapter **??**.

The universal unramified principal series module M is given by $C_c(A(\mathcal{O})N\backslash G/I)$. It is a right module over the Iwahori–Hecke algebra under convolution. Furthermore, a basis of the Iwahori–Hecke algebra is parameterised by the *affine Weyl group* $\widetilde{W}$. We may write $\widetilde{W} \cong W \ltimes \Lambda^\vee$, where $\Lambda^\vee$ is the *coroot lattice* of G. Then $\mathbb{C}[\Lambda^\vee]$ is the corresponding group algebra over $\mathbb{C}$. Then M is also a left module over $\mathbb{C}[\Lambda^\vee]$.

1. Gaussian Processes

The aim of this chapter is to review some essential mathematical machinery required for us to understand the core concepts of Gaussian Processes.

1.1. **Kernels.** Often in machine learning we are often met with the problem of how to best represent data instances as fixed size feature vectors $\boldsymbol{x}_i \in X$. For certain objects it might not be obvious at all on how to represent the data as a fixed length vector. Good examples of variable length data include textual documents and genomic data. For these types of data we can instead define a method of measuring similarity between object which does require them to be converted to a fixed length feature vector first [Mur12]. To do this we can begin by mapping the feature vectors into a Hilbert space H which provides us with an inner product $\langle \cdot, \cdot \rangle_H : H \times H \to \mathbb{R}$ and a norm $\| \cdot \|_H : H \to \mathbb{R}$. Input data is transformed into feature space vectors via a non-linear feature mapping $\Phi : X \to H$. The benefit of using a feature mapping in this way is that we can construct non-linear descision boundaries using linear models. In some instances a similarity measure can be computed directly using a function $k : X \times X$, instead of needing to construct a Φ and then computing the inner product of the transformed instances. We call such functions that act directly of our data instances kernel functions and using a kernel function to avoid computation associated with the underlying feature space is known as the kernel trick [Ste08]. These ideas are stated more formally in definition 2.

Definition 2 (Kernel)**.** *Let X be a non-empty set. Then a function $k : X \times X \to \mathbb{R}$ is called a kernel on X if there exists a Hilbert space and a map $\Phi : X \to H$ such that for all $\boldsymbol{x}, \boldsymbol{x}' \in X$ we have $k(\boldsymbol{x}, \boldsymbol{x}') = \langle \Phi(\boldsymbol{x}), \Phi(\boldsymbol{x}') \rangle_H$. We call the Φ the feature map and H the feature space of k.*

It is worth noting that almost no conditions are placed on the set X, allowing it to accommodate virtually any form of data. It is of little surprise then that neither the feature map nor the feature space are uniquely determined by the kernel. As a quick example from Ingo Steinwart and Andreas Christmann book *Support Vector Machines* [Ste08], let $X = \mathbb{R}$ and $k(x, x') = x \cdot x'$ where $x, x' \in X$. We can see that k is a kernel using the feature map $\Phi(x) \triangleq x$ and $H = \mathbb{R}$. However another suitable feature map for this particular kernel is $\Phi'(x) \triangleq \left(x/\sqrt{2}, x/\sqrt{2}\right)$ with a corresponding feature space of $H = \mathbb{R}^2$ since

$$\langle \Phi'(x), \Phi'(x') \rangle_{\mathbb{R}^2} = \frac{x'}{\sqrt{2}} \cdot \frac{x}{\sqrt{2}} + \frac{x'}{\sqrt{2}} \cdot \frac{x}{\sqrt{2}} = x \cdot x'$$

for $x, x' \in X$. While their might be numerous functions that provide some notion of similarity between data entries, these functions might not be valid kernels. Instead of needing to construct a feature map and feature space to verify that a chosen function is a valid kernel using definition 2, we can make use of a much simpler set of criteria. First, we shall need the following definition.

Definition 3 (Positive Definite and Positive Semidefinite)**.** *A function $k : K \times K \to \mathbb{R}$ is positive semidefinite if for all $n \in \mathbb{N}$ and $\alpha_1, \ldots, \alpha_n \in \mathbb{R}$ and all $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n \in X$ we have*

$$\sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j k(\boldsymbol{x}_j, \boldsymbol{x}_i) \geq 0. \tag{1}$$

Furthermore, k is said to be positive definite if for mutually distinct $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n \in X$ equality 1 only holds for $\alpha_1 = \ldots = \alpha_n = 0$ [Ste08].

Definition 4 (Symmetric). *A function $k : K \times K \to \mathbb{R}$ is called symmetric if $k(\boldsymbol{x}, \boldsymbol{x}') = k(\boldsymbol{x}', \boldsymbol{x})$ for any inputs $\boldsymbol{x}', \boldsymbol{x} \in X$* [Ste08].

Definition 5 (Gram Matrix). *For fixed $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n \in X$ the matrix $\boldsymbol{K} \in \mathbb{R}^{n \times n}$ where $\boldsymbol{K}_{i,j} \triangleq k(\boldsymbol{x}_j, \boldsymbol{x}_i)$ is the Gram matrix* [Ste08].

Note that checking if a function is positive (semi) definite is equivalent to checking that any Gram matrix produced by function is positive (semi) definite. If k is a real valued kernel corresponding to the feature map Φ, then k is symmertic by virtue of the fact that the inner product of a real Hilbert space is symmetric. Moreover k is positive definite since for $\alpha_1, \ldots, \alpha_n \in \mathbb{R}$ and $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n \in X$ we have

$$\begin{aligned}
&\sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j k(\boldsymbol{x}_j, \boldsymbol{x}_i) \\
&= \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j \langle \Phi(\boldsymbol{x}_i), \Phi(\boldsymbol{x}_j) \rangle_H \\
&= \left\| \sum_{i}^{n} \alpha_i \Phi(\boldsymbol{x}_i) \right\|_H^2 \\
&\geq 0.
\end{aligned}$$

The following theorems tell us that it is not only necessary for a kernel to be positive semi definite but it is also a sufficient condition.

Theorem 6. *A function $k : K \times K \to \mathbb{R}$ is a kernel if and only if it is symmertic and positive semidefinite* [Ste08].

1.2. **Reproducing Kernel Hilbert Spaces.** We shall now shift our attention towards reproducing kernel Hilbert spaces (RKHS) and describe their relation to kernels. We shall see that in some sense the RKHS of a kernel k is the smallest feature space for a kernel. The formal definition of a RKHS is stated in definition 7.

Definition 7 (RKHS). *Let $X \neq$ and H be a real Hilbert space over X*

(1) *A function $k : X \times X \to \mathbb{R}$ is called a reproducing kernel if we have $k(\cdot, \boldsymbol{x}) \in H$ for all $\boldsymbol{x} \in X$ and the reproducing property*

$$f(\boldsymbol{x}) = \langle f, k(\cdot, \boldsymbol{x}) \rangle$$

holds for all $f \in H$ and $x \in X$.

(2) *The space H is called a reproducing kernel Hilbert space over X if for all $\boldsymbol{x} \in X$ the Dirac functional $\delta_{\boldsymbol{x}} : H \to \mathbb{R}$ defined by $\delta_{\boldsymbol{x}}(f) \triangleq f(x),\ f \in H$ is continuous.*

[Ste08]

An important property of the RKHS is that the convergence in the norm implies pointwise convergence. Specifically in a RKHS for any sequence of functions $\{f_n\} \subset H$ where $\|f_n - f\| \to 0$ we have $|\delta_{\boldsymbol{x}}(f_n) - \delta_{\boldsymbol{x}}(f)| = |f_n(x) - f(x)| \to 0$. Note that because the evaluation function is both linear and continuous then it is also bounded in the sense that there is an $c \in \mathbb{R},\ c > 0$ such that for every $f \in H$

and a fixed $\boldsymbol{x} \in X$ we have $|\delta_{\boldsymbol{x}}(f)| \leq c\|f\|_H$ [Ber96]. This property of uniform convergence implying pointwise convergence is important since it tells us that if functions $f, g \in H$ are close in norm then their evaluation at any point is also similar. The following lemma ties together the definition of an RKHS, reproducing kernel and a kernel.

Lemma 8. *Let H be a Hilbert function space over X that has a reproducing kernel k. Then H is a RKHS and H is also a feature space of k where the feature map $\Phi : X \to H$ is given by*

$$\Phi(\boldsymbol{x}) = k(\cdot, \boldsymbol{x})$$

for some $\boldsymbol{x} \in X$. We call Φ the canonical feature map.

Proof. Since the reproducing property tells us that any Dirac functional can be represented by the reproducing kernel this means

$$|\delta_{\boldsymbol{x}}(f)| = |f(\boldsymbol{x})| = |\langle f, k(\cdot, \boldsymbol{x})\rangle| \leq \|k(\cdot, \boldsymbol{x})\|_H \cdot \|f\|_H$$

for all $\boldsymbol{x} \in X,\ f \in H$. This shows continuity of $\delta_{\boldsymbol{x}}$ for $\boldsymbol{x} \in X$. In order to show that Φ is a feature map, fix an $\boldsymbol{x}' \in X$ and set $f = k(\cdot, \boldsymbol{x}')$. Then for $\boldsymbol{x} \in X$, the reproducing property yields

$$\langle \Phi(\boldsymbol{x}'), \Phi(\boldsymbol{x})\rangle_H = \langle k\left(\cdot, \boldsymbol{x}'\right), k(\cdot, \boldsymbol{x})\rangle_H = \langle f, k(\cdot, \boldsymbol{x})\rangle_H = f(\boldsymbol{x}) = k\left(\boldsymbol{x}', \boldsymbol{x}\right).$$

□

This tells us that every Hilbert space with a reproducing kernel is a RKHS. We can also show the converse, that is, every RKHS has a unique reproducing kernel seen in theorem 9.

Theorem 9. *Let H be a RKHS over X. Then $k : X \times X \to \mathbb{R}$ defined by $k(\boldsymbol{x}', \boldsymbol{x}) = \langle \delta_{\boldsymbol{x}}, \delta_{\boldsymbol{x}'}\rangle_H,\ \boldsymbol{x}, \boldsymbol{x}' \in X$ is the only reproducing kernel of H* [Ste08].

Theorem 9 shows that a RKHS is uniquely determined by its kernel. In fact the other direction can also be shown giving a one-to-one correspondence between kernels and RKHS. This is known as the Moore-Aronszajn theorem seen in thorem 10.

Theorem 10 (Moore-Aronszajn)**.** *Suppose k is a symmertic positive definite kernel on a set X. Then there is a unique Hilbert space of functions for which k is the reproducing kernel* [Ber03].

The elements of a RKHS will often inherit the analytical properties of its corresponding kernel. This means that kernels provide a mechanism for generating spaces of functions with useful analytical properties.

1.3. **Gaussian Radial Basis Kernel.** We shall now focus on a specific class of kernel that shall be used extensively in upcoming theory and experimentation.

Definition 11 (Gaussian Radial Basis Kernel)**.** *For $d \in \mathbb{N},\ \sigma \in \mathbb{R}_{>0}$ and $\boldsymbol{z}, \boldsymbol{z}' \in \mathbb{R}^d$ we define*

$$k_\sigma\left(\boldsymbol{z}, \boldsymbol{z}'\right) \triangleq \exp\left(-\sigma^{-2}\sum_{j=1}^{d}\left(\boldsymbol{z}_j - \boldsymbol{z}'_j\right)^2\right).$$

Then k_σ is a real valued kernel called the Gaussian Radial Basis Kernel (RBF) kernel with bandwidth σ. Moreover k_σ can be computed as

$$\exp\left(\frac{-\|\boldsymbol{z}-\boldsymbol{z}'\|_2^2}{\sigma^2}\right)$$

[Ste08].

The Gaussian RBF kernel makes for a very simple an intuitive measurement of similarity between its inputs. One geometric interpretation of the Gaussian RBF kernel is that as the radius of the smallest d-sphere containing $\boldsymbol{z}, \boldsymbol{z}' \in \mathbb{R}^d$ grows the corresponding measurement of similarity decays exponentially. A visual representation of this decay is shown in figure **??**.

This kernel is infinitely differentiable meaning it has mean square derivatives of all orders and is therefore very smooth. In fact, some argue that such strong smoothness makes it unrealistic for modelling natural phenomena [Ras06, Ste99]. Nontheless, Gaussian RBF kernelis remains the one of the most widely used kernels in literature.

1.4. **Kernel Machines.** In this section, we shall look at two different machine learning models that make use of kernels to perform classification and regression. The first kernel machine we shall look at are support vector machines (SVM). SVMs where originally designed for binary classification and as such we shall only present a model for binary classification, although extensions exist that allow regression and multi-class classification.

For the binary classification problem we are tasked with labelling new samples with either one of two classes, -1 or 1. We shall assume our input space consists of vectors from $\mathbb{R}^d$ and that we provided with a labelled training set $D = \{(\boldsymbol{x}_1, y_1), (\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_n, y_n)\}$. One simple method to classify samples is by creating an affine linear hyperplane satisfying

$$\begin{aligned} \langle \boldsymbol{w}, \boldsymbol{x}_i \rangle + b > 0, \quad y_i = +1 \\ \langle \boldsymbol{w}, \boldsymbol{x}_i \rangle + b < 0, \quad y_i = -1 \end{aligned} \tag{2}$$

for some $\boldsymbol{w} \in \mathbb{R}^d$ and $b \in \mathbb{R}$ where $\|w\|_2 = 1$. Moreover we would like $\boldsymbol{w}$ and b to maximise the margin, that is the maximal distance between the separating hyperplane and the points in D. The specific $\boldsymbol{w}$ and b obtained through the training set is denoted $\boldsymbol{w}_D$ and b_D and the resulting descision function is defined as

$$f_D(\boldsymbol{x}) \triangleq \operatorname{sign}\left(\langle \boldsymbol{w}_D, \boldsymbol{x} \rangle + b_D\right).$$

There are, however, a number of short comings to this model. The most obvious is that our training data may not be linearly separable in $\mathbb{R}^d$ meaning that no such $\boldsymbol{w}_D$ and b_D exist. Moreover, when noise is introduced to the data set this model will prioritize finding a hyperplane that perfectly separates the two classes, making no comprises in misclassifying points, leaving it subject to overfitting. SVMs where introduced by Boser *et al.* [Bos92] to address the first issue of separability. Their approach was to lift the input vector into a more malleable Hilbert space H_0 using a feature map. The inputs are then classified within the new space. Unfortunately this method does nothing to address the second issue of over fitting and, if anything, actually worsens it. Cortes and Vapnik [Cor95] attempted to address this second issue

by introducing slack variables to equation 2 so that we instead need to satisfy $y_i\left(\langle \boldsymbol{w}, \Phi\left(\boldsymbol{x}_i\right)\rangle + b\right) \geq 1 - \xi_i$ for some $\xi_i \in \mathbb{R}_{>0}$. These constraints can be re-written as

$$\xi_i \geq 1 - y_i\left(\langle \boldsymbol{w}, \Phi\left(\boldsymbol{x}_i\right)\rangle + b\right)$$

and combining this this our slack constraints (that is $\xi_i \geq 0$) yields

$$\xi_i \geq \max\left\{0, 1 - y_i\left(\langle \boldsymbol{w}, \Phi\left(\boldsymbol{x}_i\right)\rangle + b\right)\right\} = L_{\text{hinge}}\left(y_i, \langle \boldsymbol{w}, \Phi\left(\boldsymbol{x}_i\right)\rangle + b\right)$$

where L_{hinge} is the hinge loss defined as

$$L_{\text{hinge}}\left(y, \eta\right) \triangleq \max\left\{0, 1 - y\eta\right\}.$$

This optimization problem can be re-written is the form

$$\min_{(\boldsymbol{w},b)\in H_0\times\mathbb{R}} \lambda\left\|\boldsymbol{w}\right\|_{H_0} + \frac{1}{n}\sum_{i=1}^{n} L_{\text{hinge}}\left(y_i, f_{(\boldsymbol{w},b)}\right)$$

where $f_{(\boldsymbol{w},b)} : X \to \mathbb{R}$ is defined as

$$f_{(\boldsymbol{w},b)} \triangleq \langle \boldsymbol{w}, \Phi\left(x_i\right)\rangle + b.$$

Unfortunately, this new embedding requires us to solve for optimal parameters in a very high, or even infinite, dimension vector space. To get around this, often the Lagrange approach is used to solve the corresponding dual problem. When the hinge loss is used the dual problem becomes

$$\begin{aligned} &\max_{\alpha\in[0,C]^n} \sum_{i=1}^{n}\alpha_i - \frac{1}{2}\sum_{i,j=1}^{n} y_i y_j \alpha_i \alpha_j \langle \Phi\left(\boldsymbol{x}_i\right), \Phi\left(\boldsymbol{x}_j\right)\rangle \\ &\text{subject to} \quad \sum_{i=1}^{n} y_i\alpha_i = 0 \end{aligned} \tag{3}$$

Notice that in the dual problem, we find that inner products are only taken with vectors that have the feature map applied to them allowing us to employ the kernel if the corresponding kernel trick described in section 1.1 is known for the feature map used so that 3 becomes

$$\begin{aligned} &\max_{\alpha\in[0,C]^n} \sum_{i=1}^{n}\alpha_i - \frac{1}{2}\sum_{i,j=1}^{n} y_i y_j \alpha_i \alpha_j k\left(\boldsymbol{x}_i, \boldsymbol{x}_j\right) \\ &\text{subject to} \quad \sum_{i=1}^{n} y_i\alpha_i = 0. \end{aligned}$$

The next machine learning model of interest that uses kernels are gaussian processes. To motivate this model, consider the time series data in figure 4.

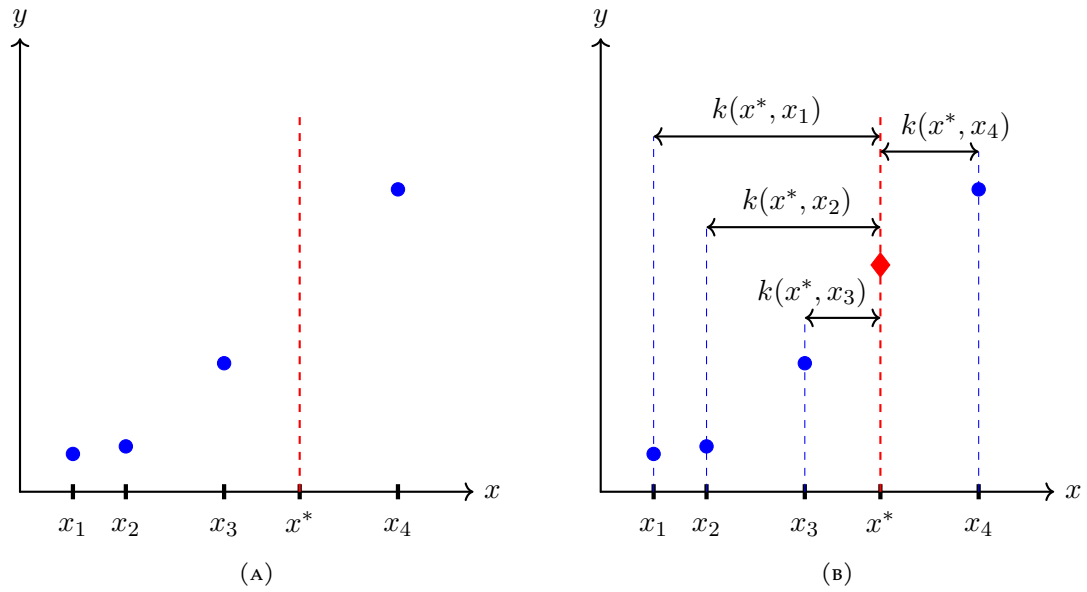
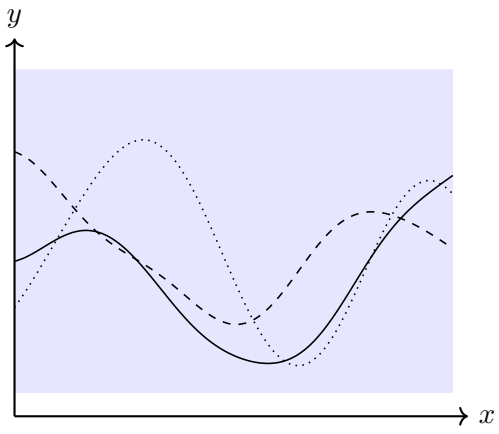
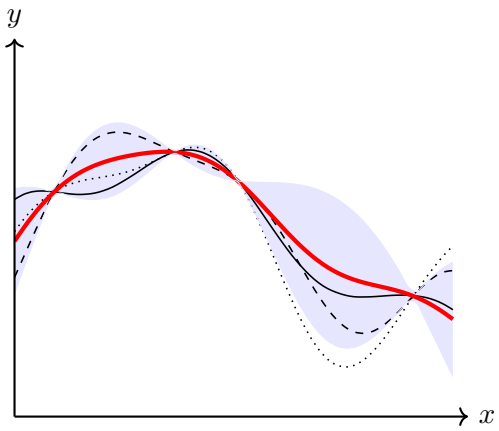


FIGURE 4. A graph of the Gaussian RBF from definition 11 for $d = 2$. Evidently, a larger value of σ slows the rate of decay increasing the similarity between the same pair of samples.





1.5. **Gaussian Processes for Regression.** A *Gaussian Process* (GP) is a collection of random variables with index set I, such that every finite subset of random variables has a joint Gaussian distribution [Ras06, Mur12].

A GP is completely characterised by a mean function $m(\boldsymbol{x})$ and a covariance function $k(\boldsymbol{x}, \boldsymbol{x'})$ on a real process as

$$m(\boldsymbol{x}) = \mathbb{E}\left[f(\boldsymbol{x})\right]$$
$$k(\boldsymbol{x}, \boldsymbol{x'}) = \mathbb{E}\left[(f(\boldsymbol{x}) - m(\boldsymbol{x}))(f(\boldsymbol{x'}) - m(\boldsymbol{x'}))\right]$$

A function $f(\boldsymbol{x})$ sampled from a GP with mean $m(\boldsymbol{x})$ and covariance $k(\boldsymbol{x}, \boldsymbol{x'})$ is written as

$$f(\boldsymbol{x}) \sim \mathcal{GP}\left(m(\boldsymbol{x}), k(\boldsymbol{x}, \boldsymbol{x'})\right)$$

Since a GP is a collection of random variables it must satisfy the consistency requirement, that is, an observation of a set of variables should not the distribution of any small sub set of the observed values. More specifically if

$$(\boldsymbol{y_1}, \boldsymbol{y_2}) \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$$

then

$$\boldsymbol{y_1} \sim \mathcal{N}(\boldsymbol{\mu_1}, \boldsymbol{\Sigma_{1,1}})$$
$$\boldsymbol{y_2} \sim \mathcal{N}(\boldsymbol{\mu_2}, \boldsymbol{\Sigma_{2,2}})$$

where $\boldsymbol{\Sigma_{1,1}}$ and $\boldsymbol{\Sigma_{2,2}}$ are the relevant sub matrices.

1.5.1. *Noise-free observations.* Typically when using GP we would like to incorporate data from observations, or training data, into our predictions on unobserved values. Let us suppose there is some obsevered data $D = \{(\boldsymbol{x}_i, \boldsymbol{f}_i) \mid i \in \{1, 2, \ldots, n\}\}$ which is (unrealistically) noise-free that we would like to model as a GP. In other words, for any sample in our dataset we can be certain that the observed value is the true value of the underlying function we wish to model. Then for the observed data

$$\boldsymbol{f} \sim \mathcal{N}\left(\boldsymbol{0}, \boldsymbol{K_{XX}}\right).$$

where $\boldsymbol{K_{XX}} = k(\boldsymbol{X}, \boldsymbol{X}) \in \mathbb{R}^{n \times n}$. We would then like to make a prediction for unobserved values say $X^* = \left[\boldsymbol{x}_1^*, \boldsymbol{x}_2^*, \ldots, \boldsymbol{x}_{n_*}^*\right]$ with value f_* as has a distribution of

$$\boldsymbol{f}_* \sim \mathcal{N}\left(\boldsymbol{0}, \boldsymbol{K_{X^*X^*}}\right).$$

where $\boldsymbol{K_{X^*X^*}} = k(\boldsymbol{X^*}, \boldsymbol{X^*}) \in \mathbb{R}^{n_* \times n_*}$. Here $\boldsymbol{f}$ and $\boldsymbol{f}_*$ are independent but we would like to give them some sort of correlation. We can do this by having them originate from the same joint distribution. According to the prior, we can write the joint distribution of the training points $\boldsymbol{f}$ and the test points $\boldsymbol{f}_*$ as

$$\begin{pmatrix} \boldsymbol{f} \\ \boldsymbol{f}_* \end{pmatrix} \sim \mathcal{N}\left(\boldsymbol{0}, \begin{pmatrix} \boldsymbol{K_{XX}} & \boldsymbol{K_{XX^*}} \\ \boldsymbol{K_{XX^*}^\intercal} & \boldsymbol{K_{X^*X^*}} \end{pmatrix}\right)$$

where $\boldsymbol{K_{XX^*}} = k(\boldsymbol{X}, \boldsymbol{X^*}) \in \mathbb{R}^{n \times n_*}$.

While the above does give us some information on $\boldsymbol{f}_*$ is related to the observed data and the test inputs, it does not provide any method to evalute $\boldsymbol{f}_*$. To do this we shall need the assistance of the following lemma

Theorem 12. (*Marginals and conditionals of an MVN* [Mur12]) *Suppose $\boldsymbol{x} = (\boldsymbol{x}_1, \boldsymbol{x}_2)$ is jointly Gaussian with parameters*

$$\boldsymbol{\mu} = \begin{pmatrix} \boldsymbol{\mu}_1 \\ \boldsymbol{\mu}_2 \end{pmatrix}, \quad \boldsymbol{\Sigma} = \begin{pmatrix} \boldsymbol{\Sigma}_{1,1} & \boldsymbol{\Sigma}_{1,2} \\ \boldsymbol{\Sigma}_{2,1} & \boldsymbol{\Sigma}_{2,2} \end{pmatrix}$$

then the posterior conditional is given by

$$\begin{aligned} \boldsymbol{x}_2 \mid \boldsymbol{x}_1 &\sim \mathcal{N}\left(\boldsymbol{x}_2 \mid \boldsymbol{\mu}_{2|1}, \boldsymbol{\Sigma}_{2|1}\right) \\ \boldsymbol{\mu}_{2|1} &= \boldsymbol{\mu}_2 + \boldsymbol{\Sigma}_{2,1}\boldsymbol{\Sigma}_{1,1}^{-1}\left(\boldsymbol{x}_1 - \boldsymbol{\mu}_1\right) \\ \boldsymbol{\Sigma}_{2|1} &= \boldsymbol{\Sigma}_{2,2} - \boldsymbol{\Sigma}_{2,1}\boldsymbol{\Sigma}_{1,1}^{-1}\boldsymbol{\Sigma}_{1,2} \end{aligned}$$

Thus finding a mean an covariance for $\boldsymbol{f}_*$ requires a direct application of Theorem 12 which gives

$$\boldsymbol{f}_* \mid \boldsymbol{K_{XX^*}}, \boldsymbol{K_{XX}}, \boldsymbol{f} \sim \mathcal{N}\left(\boldsymbol{\mu}^*, \boldsymbol{\Sigma}^*\right)$$

where

$$\begin{aligned} \boldsymbol{\mu}^* &= \boldsymbol{0} + \boldsymbol{K_{XX^*}^\intercal}\boldsymbol{K_{XX}^{-1}}\left(\boldsymbol{f} - \boldsymbol{0}\right) \\ &= \boldsymbol{K_{XX^*}^\intercal}\boldsymbol{K_{XX}^{-1}}\boldsymbol{f} \end{aligned}$$

and

$$\boldsymbol{\Sigma}^* = \boldsymbol{K_{X^*X^*}} - \boldsymbol{K_{XX^*}^\intercal}\boldsymbol{K_{XX}^{-1}}\boldsymbol{K_{XX^*}}$$

meaning we can write a distribution for $\boldsymbol{f}_*$ as

$$\boldsymbol{f}_* \mid \boldsymbol{K_{XX^*}}, \boldsymbol{K_{XX}}, \boldsymbol{f} \sim \mathcal{N}\left(\boldsymbol{K_{XX^*}^\intercal}\boldsymbol{K_{XX}^{-1}}\boldsymbol{f}, \boldsymbol{K_{X^*X^*}} - \boldsymbol{K_{XX^*}^\intercal}\boldsymbol{K_{XX}^{-1}}\boldsymbol{K_{XX^*}}\right) \tag{4}$$

Function values from the unobserved inputs $\boldsymbol{X^*}$ can be estimated using the mean of $\boldsymbol{f}_*$ evaluted in 4.

1.5.2. *Prediction with Noisy observations.* When attempting to model our value function we usually do not have access to the value function itself but a noisy version thereof, $y = f(\boldsymbol{x}) + \varepsilon$ where $\varepsilon \mathcal{N}(0, \sigma_n^2)$ meaning the prior on the noisy values becomes

$$\operatorname{cov}(\boldsymbol{y}) = \boldsymbol{K_{XX}} + \sigma_n^2 \boldsymbol{I}$$

The reason why noise is only added along the diagonal follows from the assumption of independence in our data. We can write out the new distribution of the observed noisy values along the points at which we wish to test the underlying function as

$$\begin{pmatrix} \boldsymbol{f} \\ \boldsymbol{f}_* \end{pmatrix} \sim \mathcal{N}\left(\boldsymbol{0}, \quad \begin{pmatrix} \boldsymbol{K_{XX}} + \sigma_n^2 \boldsymbol{I} & \boldsymbol{K_{XX^*}} \\ \boldsymbol{K_{XX^*}^\intercal} & \boldsymbol{K_{X^*X^*}} \end{pmatrix}\right)$$

Using a similar we arrive at a similar condition distribution of $\boldsymbol{f}_* \mid \boldsymbol{K_{XX^*}}, \boldsymbol{K_{XX}}, \boldsymbol{f}$ we arrive at one of the most fundamental equations for GP regression tasks

$$\begin{aligned} \boldsymbol{f}_* \mid \boldsymbol{K_{XX^*}}, \boldsymbol{K_{XX}}, \boldsymbol{y} \sim & \mathcal{N}\left(\overline{\boldsymbol{f}_*}, \operatorname{cov}(\boldsymbol{f}_*)\right) \\ & \overline{\boldsymbol{f}_*} := \boldsymbol{K_{XX^*}^\intercal}\left[\boldsymbol{K_{XX}} + \sigma_n^2 \boldsymbol{I}\right]^{-1} \boldsymbol{y} \\ & \operatorname{cov}(\boldsymbol{f}_*) = \boldsymbol{K_{X^*X^*}} - \boldsymbol{K_{XX^*}^\intercal}\left[\boldsymbol{K_{XX}} + \sigma_n^2 \boldsymbol{I}\right]^{-1} \boldsymbol{K_{XX^*}} \end{aligned}$$

2. Krylov Subspace Methods

In this section we will focus on how iterative methods, in particular a class of iterative methods called Krylov Subspace methods, may be used to solve a linear system $\boldsymbol{Ax} = \boldsymbol{b}$. While non-iterative methods exist to solve such systems virtually all of them carry an unwieldy runtime of $\mathcal{O}\left(n^3\right)$ for a system of n parameters. Even for current computer systems, this renders many common matrix problems untractable. Consequently the focus of solving linear systems has shifted towards iterative methods. While iterative methods typically demand certain structural properties of the matrices, such as symmetry and positive definiteness, this generally is not a problem since the majority of large matrix problems that, by mature, endow these systems with the desired properties. For example, in the context of this paper the Gram matrices used to solve linear systems in Gaussian Processes possess both symmetry and positive definiteness. There are also a number of other properties of iterative methods which make them rather attractive to users. To start, iterative Krylov subspace methods are guranteed to converge to an exact solution within a finite number of iterations and even if the method is prematurely stopped before reaching an exact solution, the approximation obtained on the final iteration will in some sense be a good enough estimate of our exact solution. Furthermore, unlike most non-iterative methods, Krylov subspace methods do not require an explicit form of the matrix $\boldsymbol{A}$ and instead only requires some routine or process for computing $\boldsymbol{Ax}$.

2.1. Krylov Subspaces.

We will motivate the Krylov subspaces by observing their usefullness in solving linear systems. To this end, consider the problem of solving the linear system

$$\boldsymbol{Ax}^\star = \boldsymbol{b} \tag{5}$$

where no explicit form of $\boldsymbol{A}$ is available and instead one must draw information from $\boldsymbol{A}$ solely through a routine that can evaluate $\boldsymbol{Av}$ for any $\boldsymbol{v}$. How could this routine be utilized in such a manner to provide with a solution to equation 5? Before answering this, consider the following theorem

Theorem 13. *For $\boldsymbol{A} \in \mathbb{K}^{n\times n}$ if $\|\boldsymbol{A}\| = q < 1$ then $\mathbb{1} - \boldsymbol{A}$ is invertible and its inverse admits the following representation*

$$(\mathbb{1} - \boldsymbol{A})^{-1} = \sum_{k=0}^{\infty} \boldsymbol{A}^k.$$

[Ber96]

Consider a matrix for which $\|\boldsymbol{A}\| < 2$, it follows that $\|\mathbb{1} - \boldsymbol{A}\| < 1$ meaning $\mathbb{1} - (\mathbb{1} - \boldsymbol{A})$ is invertible and $\boldsymbol{A}^{-1} = (\mathbb{1} - (\mathbb{1} - \boldsymbol{A}))^{-1} = \sum_{k=0}^{\infty} (\mathbb{1} - \boldsymbol{A})^k$. Thinking back to equation 5 for any $x_0 \in \mathbb{K}^n$ we have

$$\begin{aligned}
\boldsymbol{x}^\star &= \boldsymbol{A}^{-1}\boldsymbol{b} = \boldsymbol{A}^{-1}\left(\boldsymbol{Ax}^\star - \boldsymbol{Ax_0} + \boldsymbol{Ax_0}\right) \\
&= \boldsymbol{x_0} + \boldsymbol{A}^{-1}\boldsymbol{r_0} \\
&= \boldsymbol{x_0} + \sum_{k=0}^{\infty} (\mathbb{1} - \boldsymbol{A})^k
\end{aligned}$$

where $\boldsymbol{r_0} = \boldsymbol{Ax}^\star - \boldsymbol{Ax_0}$. A natural question that arises is that can we find a closed form solution of the above equation? To answer this question we need to enlist the help of the Cayley-Hamilton theorem.

Theorem 14 (Cayley-Hamilton)**.** *Let $p_n(\lambda) = \sum_{i=0}^{n} c_i \lambda^i$ be the characteristic polynomial of the matrix $\boldsymbol{A} \in \mathbb{K}^{n\times n}$, then $p_n(\boldsymbol{A}) = \boldsymbol{0}$.* ***THIS NEEDS A CITATION***

The Cayley-Hamilton theorem implies that

$$\begin{aligned} 0 &= c_0 + c_1\boldsymbol{A} + \ldots + c_{n-1}\boldsymbol{A}^{n-1} + c_n\boldsymbol{A}^n \\ 0 &= \boldsymbol{A}^{-1}c_0 + c_1 + \ldots + c_{n-1}\boldsymbol{A}^{n-2} + c_n\boldsymbol{A}^{n-1} \\ \boldsymbol{A}^{-1} &= \alpha_0 + c_1 + \ldots + \alpha_{n-1}\boldsymbol{A}^{n-2} + \alpha_n\boldsymbol{A}^{n-1} \end{aligned}$$

where $\alpha_i = -c_i/c_0$. This demonstrates that $\boldsymbol{A}^{-1}$ can be represented as a matrix polynomial of degree $n-1$. This means that $\sum_{k=0}^{\infty}(\mathbb{1}-\boldsymbol{A})^k$ indeed possess a closed form solution namely

$$\boldsymbol{x}^\star = \boldsymbol{x_0} + \boldsymbol{A}^{-1}\boldsymbol{r_0} = \alpha_0 + c_1 + \ldots + \alpha_{n-1}\boldsymbol{A}^{n-2} + \alpha_n\boldsymbol{A}^{n-1}.$$

This also shows that $\boldsymbol{x}^\star \in \text{l.s}\left\{\boldsymbol{r_0}, \boldsymbol{A}\boldsymbol{r_0}, \boldsymbol{A}^2\boldsymbol{r_0}, \ldots, \boldsymbol{A}^{n-1}\boldsymbol{r_0}\right\}$. One idea for finding a solution to equation 5 is to use our routine for evaluting $\boldsymbol{A}\boldsymbol{v}$ to iteratively compute new basis elements for the space generated by $\left\{\boldsymbol{r_0}, \boldsymbol{A}\boldsymbol{r_0}, \boldsymbol{A}^2\boldsymbol{r_0}, \ldots, \boldsymbol{A}^{n-1}\boldsymbol{r_0}\right\}$ and at each step carefully choosing a $\boldsymbol{x_k}$ such that $\boldsymbol{x_k}$ approaches $\boldsymbol{x}^\star$, in some form. The subspace constructed using this technique is so important that is has its own name.

Definition 15 (Krylov Subspace)**.** *The Krylov Subspace of order k generated by the matrix $\boldsymbol{A} \in \mathbb{K}^{n\times n}$ and the vector $\boldsymbol{v} \in \mathbb{K}$ is defined as*

$$\mathcal{K}_k(\boldsymbol{A}, \boldsymbol{v}) = \text{l.s}\left\{\boldsymbol{r_0}, \boldsymbol{A}\boldsymbol{r_0}, \boldsymbol{A}^2\boldsymbol{r_0}, \ldots, \boldsymbol{A}^{n-1}\boldsymbol{r_0}\right\}$$

for $k \geq 1$ and $\mathcal{K}_k(\boldsymbol{A}, \boldsymbol{v}) = \{\boldsymbol{0}\}$.

For the purposes of solving equation 5 it is of much interest to understand how $\mathcal{K}_k(\boldsymbol{A}, \boldsymbol{v})$ grows for larger and larger k since a solution for equation 5 will be present in a Krylov Subspace that cannot be grown any larger. In other words, an exact solution can be constructed once we have extracted all the information from $\boldsymbol{A}$ through multiplication of $\boldsymbol{r_0}$. The following theorem provides information on how exactly the Krylov Subspace grows as k increases.

Theorem 16. *There is a positive called the grade of $\boldsymbol{v}$ with respect to $\boldsymbol{A}$, denoted $t_{\boldsymbol{v},\boldsymbol{A}}$, where*

$$\dim\left(\mathcal{K}_k(\boldsymbol{A}, \boldsymbol{v})\right) = \begin{cases} k, & k \leq t \\ t, & k \geq t \end{cases}$$

Theorem 16 essentially tells us that for $k \leq t_{\boldsymbol{v},\boldsymbol{A}}$ that $\boldsymbol{A}^k\boldsymbol{v}$ is linearly independent to $\boldsymbol{A}^i\boldsymbol{v}$ for $0 \leq i \leq k-1$ meaning $\left\{\boldsymbol{v}, \boldsymbol{A}\boldsymbol{v}, \boldsymbol{A}^2\boldsymbol{v}, \ldots, \boldsymbol{A}^{n-1}\boldsymbol{v}\right\}$ serves as a basis for $\mathcal{K}_k(\boldsymbol{A}, \boldsymbol{v})$ and that $\mathcal{K}_{k-1}(\boldsymbol{A}, \boldsymbol{v}) \subsetneq \mathcal{K}_k(\boldsymbol{A}, \boldsymbol{v})$. Conversely, any new vectors formed beyond $t_{\boldsymbol{v},\boldsymbol{A}}$ will be linearly independent meaning $\mathcal{K}_k(\boldsymbol{A}, \boldsymbol{v}) \subsetneq \mathcal{K}_{k+1}(\boldsymbol{A}, \boldsymbol{v})$ for $k \geq t_{\boldsymbol{v},\boldsymbol{A}}$. While $t_{\boldsymbol{v},\boldsymbol{A}}$ clearly plays a role in determining a suitable basis for which $\boldsymbol{A}^{-1}\boldsymbol{b}$ lies in its importance is made abundantly clear in the following corollary.

Corollary 17.

$$t_{\boldsymbol{v},\boldsymbol{A}} = \min\left\{k \mid \boldsymbol{A}^{-1}\boldsymbol{v} \in \mathcal{K}_k(\boldsymbol{A}, \boldsymbol{v})\right\}$$

Proof. Recall from Cayley-Hamilton (theorem 14) that

$$\boldsymbol{A}^{-1}\boldsymbol{v} = \sum_{i=0}^{n-1} \alpha_i \boldsymbol{A}^i \boldsymbol{v}$$

But since $\mathcal{K}_k(\boldsymbol{A}, \boldsymbol{v}) = \mathcal{K}_{k+1}(\boldsymbol{A}, \boldsymbol{v})$ for $k \geq t_{\boldsymbol{v},\boldsymbol{A}}$

$$\boldsymbol{A}^{-1}\boldsymbol{v} = \sum_{i=0}^{t-1} \beta_i \boldsymbol{A}^i \boldsymbol{v}$$

meaing $\boldsymbol{A}^{-1}\boldsymbol{v} \in \mathcal{K}_k(\boldsymbol{A}, \boldsymbol{v})$ for $k \geq t_{\boldsymbol{v},\boldsymbol{A}}$. Suppose for the sake of contradiction that this also holds for $k = t_{\boldsymbol{v},\boldsymbol{A}} - 1$, that is, $\boldsymbol{A}^{-1}\boldsymbol{v} = \sum_{i=0}^{t-2} \gamma_i \boldsymbol{A}^i \boldsymbol{v}$. However, this gives

$$\boldsymbol{v} = \sum_{i=0}^{t-2} \gamma_i \boldsymbol{A}^{i+1}\boldsymbol{v} = \sum_{i=0}^{t-1} \gamma_{i-1} \boldsymbol{A}^i \boldsymbol{v}$$

implying $\left\{\boldsymbol{v}, \boldsymbol{A}\boldsymbol{v}, \boldsymbol{A}^2\boldsymbol{v}, \ldots, \boldsymbol{A}^{t-1}\boldsymbol{v}\right\}$ are linearly dependent which means that $\dim\left(\mathcal{K}_k(\boldsymbol{A}, \boldsymbol{v})\right) < t$, which provides us with our contrdiction. □

This machinery allows us to make a much stronger statement on the where abouts of $\boldsymbol{x}^\star$ in relation to the Krylov Subspaces.

Corollary 18. *For any $\boldsymbol{x_0}$, we have*

$$\boldsymbol{x}^\star \in \boldsymbol{x_0} + \mathcal{K}_{t_{\boldsymbol{r_0},\boldsymbol{A}}}(\boldsymbol{A}, \boldsymbol{r_0})$$

where $\boldsymbol{r_0} = \boldsymbol{b} - \boldsymbol{A}\boldsymbol{x_0}$.

2.2. **Gram-Schmidt Process and QR factorisations.** Many areas of linear algebra involving studing the column space of matrices. The QR factorisation provides us with a powerful tool to better understand the column space of a matrix as well as serving as an important factorisation mechanism for many numerical methods. Suppose that a matrix $\boldsymbol{A} = [\boldsymbol{a}_1, \boldsymbol{a}_2, \ldots, \boldsymbol{a}_n] \in \mathbb{K}^{n\times n}$ has full rank. The idea of a QR factorisation is to find an alternative orthornormal basis for $(\boldsymbol{a}_i)_{i=1}^n$, say $(\boldsymbol{q}_i)_{i=1}^n$, and to somehow relate the original matrix $\boldsymbol{A}$ to a new matrix whose columns are $(\boldsymbol{q}_i)_{i=1}^n$. Consider the following procedure that allows us to find an orthornormal basis $(\boldsymbol{q}_i)_{i=1}^n$ for which $\mathrm{l.s}\{(\boldsymbol{a}_i)_{i=1}^n\} = \mathrm{l.s}\{(\boldsymbol{q}_i)_{i=1}^n\}$. First set $\boldsymbol{q}_1 = \frac{\boldsymbol{a}_1}{\|\boldsymbol{a}_i\|}$, clearly $\mathrm{l.s}\{\boldsymbol{a}_1\} = \mathrm{l.s}\{\boldsymbol{q}_1\}$. Next, construct a vector $\boldsymbol{q}_2' = \boldsymbol{a}_2 - r_{1,2}\cdot\boldsymbol{q}_1$ so that $\boldsymbol{q}_2' \perp \boldsymbol{q}_1$. This means

$$\begin{aligned} 0 &= \langle \boldsymbol{q}_1, \boldsymbol{q}_2' \rangle \\ 0 &= \langle \boldsymbol{q}_1, \boldsymbol{a}_2 - r_{1,2}\cdot\boldsymbol{q}_1 \rangle \\ 0 &= \langle \boldsymbol{q}_1, \boldsymbol{a}_2 \rangle - r_{1,2}\cdot\langle \boldsymbol{q}_1, \boldsymbol{q}_1 \rangle \\ r_{1,2} &= \langle \boldsymbol{q}_1, \boldsymbol{a}_2 \rangle \end{aligned}$$

Since $\boldsymbol{q}_2'$ may not be a unit vector we set $\boldsymbol{q}_2 = \frac{\boldsymbol{q}_2'}{\|\boldsymbol{q}_2'\|}$ where $\mathrm{l.s}(\{\boldsymbol{a}_1, \boldsymbol{a}_2\}) = \mathrm{l.s}(\{\boldsymbol{q}_1, \boldsymbol{q}_2\})$. Continuing the vector $\boldsymbol{q}_3'$ is constructed so that

$$\boldsymbol{q}_3' = \boldsymbol{a}_3 - \boldsymbol{r}_{1,3}\boldsymbol{q}_1 - \boldsymbol{r}_{2,3}\boldsymbol{q}_2$$

are chosen so that $\boldsymbol{q}_3'$ is orthogonal to both $\boldsymbol{q}_2$ and $\boldsymbol{q}_1$. This amounts to setting $r_{1,3} = \langle \boldsymbol{q}_1, \boldsymbol{a}_3 \rangle$ and $r_{2,3} = \langle \boldsymbol{q}_2, \boldsymbol{a}_3 \rangle$. Similarly, $\boldsymbol{q}_3'$ is normalized so that $\boldsymbol{q}_3 = \frac{\boldsymbol{q}_3'}{\|\boldsymbol{q}_3'\|}$ and $\mathrm{l.s}(\{\boldsymbol{a}_1, \boldsymbol{a}_2, \boldsymbol{a}_3\}) = \mathrm{l.s}(\{\boldsymbol{q}_1, \boldsymbol{q}_2, \boldsymbol{q}_3\})$.

Continuing in this fashion the k^{th} vector in our orthornormal basis is computed as

$$\boldsymbol{q}_k = \frac{\boldsymbol{a}_k - \sum_{i=1}^{k-1} r_{i,k} \cdot \boldsymbol{q}_i}{r_{k,k}} \tag{6}$$

where $r_{i,k} = \langle \boldsymbol{q}_i, \boldsymbol{a}_k \rangle$, $r_{k,k} = \|\boldsymbol{a}_k - \sum_{i=1}^{k-1} r_{i,k} \cdot \boldsymbol{q}_i\|$ and $\mathrm{l.s}\left(\{\boldsymbol{a}_1, \boldsymbol{a}_2, \ldots, \boldsymbol{a}_k\}\right) = \mathrm{l.s}\left(\{\boldsymbol{q}_1, \boldsymbol{q}_2, \ldots, \boldsymbol{q}_k\}\right)$. This procedure is famiously known as the Gram-Schmidt process [Ber96, Tre97, Dem97] and is summarized in the following algorithm.

Algorithm 1: Classical Gram-Schmidt

input : A basis $(\boldsymbol{a}_i)_{i=1}^n$.
output: An orthornormal basis $(\boldsymbol{q}_i)_{i=1}^n$ such that $\mathrm{l.s}\left\{(\boldsymbol{a}_i)_{i=1}^n\right\} = \mathrm{l.s}\left\{(\boldsymbol{q}_i)_{i=1}^n\right\}$

for $k = 1$ **to** n **do**
  $\boldsymbol{q}'_k = \boldsymbol{a}_k$
  for $i = 1$ **to** $k-1$ **do**
    $r_{i,k} = \langle \boldsymbol{q}_i, \boldsymbol{a}_k \rangle$
    $\boldsymbol{q}'_k = \boldsymbol{q}'_k - r_{i,k}\boldsymbol{q}_i$
  end
  $r_{k,k} = \|\boldsymbol{q}'_k\|$
  $\boldsymbol{q}_k = \boldsymbol{q}'_k / r_{k,k}$
end
return $(\boldsymbol{q}_i)_{i=1}^n$

Relating the column space of $\boldsymbol{A}$ to the orthornormal basis $(\boldsymbol{q}_i)_{i=1}^n$ in a matrix form

$$[\boldsymbol{a}_1, \boldsymbol{a}_2, \ldots \boldsymbol{a}_n] = [\boldsymbol{q}_1, \boldsymbol{q}_2, \ldots \boldsymbol{q}_n] \begin{bmatrix} r_{1,1} & r_{1,2} & \cdots & r_{1,n} \\ & r_{2,2} & & \vdots \\ & & \ddots & \vdots \\ & & & r_{n,n} \end{bmatrix}$$

or more succinctly

$$\boldsymbol{A} = \boldsymbol{Q}\boldsymbol{R} \tag{7}$$

where $\boldsymbol{Q} = [\boldsymbol{q}_1, \boldsymbol{q}_2, \ldots \boldsymbol{q}_n]$ and $(\boldsymbol{R})_{i,j} = r_{i,j}$ for $i \leq j$ and $(\boldsymbol{R})_{i,j} = 0$ for $i > j$. This is exactly the QR factorisation for a full rank matrix. Note that $\mathrm{Range}\,(\boldsymbol{A}) = \mathrm{Range}\,(\boldsymbol{Q})$. In general, any square matrix $\boldsymbol{A} \in \mathbb{K}^{m \times n}$ may be decomposed as $\boldsymbol{A} = \boldsymbol{Q}\boldsymbol{R}$ where $\boldsymbol{Q} \in \mathbb{K}^{m \times m}$ is an orthogonal matrix and $\boldsymbol{R} \in \mathbb{K}^{m \times n}$ is an upper triangular matrix. This is known as a full QR factorisation. Since bottom $(m-n)$ rows of this $\boldsymbol{R}$ consists entirely of zeros, it is often useful to partition the full QR factorisation in the following manner to shed vacuous entries

$$\boldsymbol{A} = \boldsymbol{Q}\boldsymbol{R} = \boldsymbol{Q}\begin{bmatrix} \hat{\boldsymbol{R}} \\ \boldsymbol{0}_{(m-n)\times n} \end{bmatrix} = \begin{bmatrix} \hat{\boldsymbol{Q}} & \boldsymbol{Q}' \end{bmatrix} \begin{bmatrix} \hat{\boldsymbol{R}} \\ \boldsymbol{0}_{(m-n)\times n} \end{bmatrix} = \hat{\boldsymbol{Q}}\hat{\boldsymbol{R}}.$$

This alternate decomposition is called the reduced (or somtimes the thin) QR factorization. We shall state the following two theorems on the QR factorization are stated without proof.

Theorem 19. *Every* $\boldsymbol{A} \in \mathbb{K}^{m\times n},\ (m \geq n)$ *has a full* QR *factorisation, hence also a reduced* QR *factorisation.* [Tre97]

Theorem 20. *Each* $\boldsymbol{A} \in \mathbb{K}^{m\times n},\ (m \geq n)$ *of full rank has a unique reduced* QR *factorisation* $\boldsymbol{A} = \hat{\boldsymbol{Q}}\hat{\boldsymbol{R}}$ *with* $r_{k,k} > 0$. [Tre97]

In practice the classical Gram-Schmidt process described in algorithm 1 is rarely used as the procedure becomes numerically unstable if $(\boldsymbol{a}_i)_{i=1}^n$ are almost linearly dependent. Before looking for ways to resolve these numerical instabilities a quick recap of projectors has been devised. A square matrix $\boldsymbol{P}_G$ acting on a Hilbert space H that sends $\boldsymbol{x} \in H$ to its projection onto a subspace G is called the projector onto G. If $(\boldsymbol{q}_k)_{k=1}^m$ is an orthornormal basis in G then

$$\boldsymbol{P}_G = \boldsymbol{Q}\boldsymbol{Q}^*$$

where $\boldsymbol{Q} = [\boldsymbol{q}_1, \boldsymbol{q}_2, \dots \boldsymbol{q}_m, 0, \dots, 0] \in \mathbb{K}^{n\times n}$. A special class of projectors which isolates the components of a given vector onto a one dimensional subspace spanned by a single unit vector $\boldsymbol{q}$ called a rank one orthogonal projector, denoted as $\boldsymbol{P}_q$. Each k in the classical Gram-Schmidt process $\boldsymbol{q}'_k$ using the following orthogonal projection

$$\boldsymbol{q}'_k = \boldsymbol{P}_{A_k^\perp}\boldsymbol{a}_k \tag{8}$$

where $A_k = \mathrm{l.s}\left\{\boldsymbol{a}_i\right\}_{i=1}^k$ and $\boldsymbol{P}_{A_1^\perp} = \mathbb{1}$ for convenience. A modified version of the Gram-Schmidt process performs the same orthogonal projection broken up as $k-1$ orthogonal projections of rank $n-1$ as so

$$\begin{aligned}
\boldsymbol{q}'_k &= \boldsymbol{P}_{A_k^\perp}\boldsymbol{a}_k \\
&= \left(\mathbb{1} - \boldsymbol{Q}_k\boldsymbol{Q}_k^*\right)\boldsymbol{a}_k \\
&= \left(\prod_{i=1}^{k-1}\left(\mathbb{1} - \boldsymbol{q}_i\boldsymbol{q}_i^*\right)\right)\boldsymbol{a}_k \\
&= \left(\mathbb{1} - \boldsymbol{q}_1\boldsymbol{q}_1^*\right)\left(\mathbb{1} - \boldsymbol{q}_1\boldsymbol{q}_1^*\right)\cdots\left(\mathbb{1} - \boldsymbol{q}_{k-1}\boldsymbol{q}_{k-1}^*\right)\boldsymbol{a}_k \\
&= \boldsymbol{P}_{\boldsymbol{q}_k^\perp}\cdots\boldsymbol{P}_{\boldsymbol{q}_1^\perp}\boldsymbol{a}_k
\end{aligned}$$

While its clear that $\boldsymbol{P}_{A_k^\perp}\boldsymbol{a}$ and $\boldsymbol{P}_{\boldsymbol{q}_k^\perp}\cdots\boldsymbol{P}_{\boldsymbol{q}_1^\perp}\boldsymbol{a}_k$ used for computing $\boldsymbol{q}'_k$ are algebraically, they differ arithmetically as the latter expression evaluates $\boldsymbol{q}'_k$ using the follow procedure

$$\begin{aligned}
\boldsymbol{q}_k^{(1)} &= \boldsymbol{a}_k \\
\boldsymbol{q}_k^{(2)} &= \boldsymbol{P}_{\boldsymbol{q}_1^\perp}\boldsymbol{q}_k^{(1)} \\
\boldsymbol{q}_k^{(3)} &= \boldsymbol{P}_{\boldsymbol{q}_2^\perp}\boldsymbol{q}_k^{(2)} \\
&\vdots \\
\boldsymbol{q}'_k = \boldsymbol{q}_k^{(k)} &= \boldsymbol{P}_{\boldsymbol{q}_{k-1}^\perp}\boldsymbol{q}_k^{(k-1)}
\end{aligned}$$

Applying projections sequentially in this manner produces smaller numerical errors. The modified Gram-Schmidt process [Tre97, Dem97] is summarized in the following algorithm.

Algorithm 2: Modified Gram-Schmidt

input : A basis $\{\boldsymbol{a}_i\}_{i=1}^n$.
output: An orthornormal basis $\{\boldsymbol{q}_i\}_{i=1}^n$ such that $\mathrm{l.s}\{\boldsymbol{a}_i\}_{i=1}^n = \mathrm{l.s}\{\boldsymbol{q}_i\}_{i=1}^n$

for $k = 1$ **to** n **do**
  $\boldsymbol{q}'_k = \boldsymbol{a}_k$
end
for $k = 1$ **to** n **do**
  $r_{k,k} = \|\boldsymbol{q}'_k\|$
  $\boldsymbol{q}_k = \boldsymbol{q}'_k / r_{k,k}$
  for $i = k+1$ **to** n **do**
    $r_{i,k} = \langle \boldsymbol{q}_k, \boldsymbol{q}'_i \rangle$
    $\boldsymbol{q}_i = \boldsymbol{q}_i - r_{i,k}\boldsymbol{q}_i$
  end
end
return $\{\boldsymbol{q}_i\}_{i=1}^n$

2.3. **Arnoldi and Lanczos Algorithm.** As a quick reminder, we are in search of an iterative process to solve the linear system $\boldsymbol{A}\boldsymbol{x}^\star = \boldsymbol{b}$ where no explicit form of $\boldsymbol{A}$ is available and we may only rely on a routine that computes $\boldsymbol{A}\boldsymbol{v}$ for any $\boldsymbol{v}$ to extract information on $\boldsymbol{A}$. In section 2.1 we discovered that $\boldsymbol{x}^\star \in \mathcal{K}_{t_{\boldsymbol{r}_0,\boldsymbol{A}}}(\boldsymbol{A}, \boldsymbol{r}_0)$. With many iterative methods, computing an exact value for $\boldsymbol{x}^\star$ is out the question with the view that $t_{\boldsymbol{r}_0,\boldsymbol{A}}$ is impractically large. We must instead resort to approximating $\boldsymbol{x}^\star$ by $\boldsymbol{x}_k$ for which $\boldsymbol{x}^k \in \mathcal{K}_k(\boldsymbol{A}, \boldsymbol{r}_0)$ where $k \ll t_{\boldsymbol{r}_0}$. To find an appropriate value for $\boldsymbol{x}_k$, a good start would be to find a basis $\mathcal{K}_k(\boldsymbol{A}, \boldsymbol{r}_0)$. Definition 15 showed us that $\left\{\boldsymbol{A}^{i-1}\boldsymbol{r}_0\right\}_{i=1}^k$ serves as a basis for $\mathcal{K}_k(\boldsymbol{A}, \boldsymbol{r}_0)$. However, for numericcal reasons this is a poor choice of basis since this each consecutive term becomes closer and closer to being linearly dependent. To search for a more appriporate basis, set $n = t_{\boldsymbol{r}_0,\boldsymbol{A}}$ so that $\boldsymbol{x}^\star \in \mathcal{K}_n(\boldsymbol{A}, \boldsymbol{r}_0)$. Let $\boldsymbol{K} \in \mathbb{K}^{n\times n}$ be the invertible matrix

$$\boldsymbol{K} = \left[\boldsymbol{r}_0, \boldsymbol{A}\boldsymbol{r}_0, \ldots, \boldsymbol{A}^{n-1}\boldsymbol{r}_0\right].$$

Since $\boldsymbol{K}$ is invertible we can compute $\boldsymbol{c} = -\boldsymbol{K}^{-1}\boldsymbol{A}^n\boldsymbol{r}_0$ so that

$$\boldsymbol{A}\boldsymbol{K} = \left[\boldsymbol{A}\boldsymbol{r}_0, \boldsymbol{A}^2\boldsymbol{r}_0, \ldots, \boldsymbol{A}^n\boldsymbol{r}_0\right]$$
$$\boldsymbol{A}\boldsymbol{K} = \boldsymbol{K}\cdot[\boldsymbol{e}_2, \boldsymbol{e}_3, \ldots, \boldsymbol{e}_n, -\boldsymbol{c}] \triangleq \boldsymbol{K}\boldsymbol{C}$$

or, in a more verbose form

$$\boldsymbol{K}^{-1}\boldsymbol{A}\boldsymbol{K} = \boldsymbol{C} = \begin{bmatrix} 0 & 0 & \cdots & 0 & -c_1 \\ 1 & 0 & \cdots & 0 & -c_2 \\ 0 & 1 & \cdots & 0 & \vdots \\ \vdots & \vdots & \cdots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & -c_n \end{bmatrix}.$$

Note here that $\boldsymbol{C}$ is upper Hessenberg. While this form is simple, it is of little practical use since the matrix $\boldsymbol{K}$ is very likely to be ill-conditioned. To remedy this we can replace $\boldsymbol{K}$ with an orthogonal matrix which

spans the same space. These are exactly the properties that the $\boldsymbol{Q}$ matrix offers in the QR-factorisation of $\boldsymbol{K}$. With this in mind let $\boldsymbol{K} = \boldsymbol{QR}$ be the full QR-factorisation of $\boldsymbol{K}$. Then

$$\begin{aligned}\boldsymbol{AQR} &= \boldsymbol{AK}\\ \boldsymbol{AQ} &= \boldsymbol{AKR}^{-1}\\ \boldsymbol{AQ} &= \boldsymbol{KCR}^{-1}\\ \boldsymbol{AQ} &= \boldsymbol{QRCR}^{-1}\\ \boldsymbol{AQ} &\triangleq \boldsymbol{QH}.\end{aligned}$$

Since $\boldsymbol{R}$ and $\boldsymbol{R}^{-1}$ and both upper triangular and $\boldsymbol{C}$ is upper Hessenberg, $\boldsymbol{H}$ is also upper Hessenberg. This form provides us with a $\boldsymbol{Q}$ such that the range of $\boldsymbol{Q}$ is $\mathcal{K}_n(\boldsymbol{A}, \boldsymbol{r}_0)$ and

$$\boldsymbol{Q}^\intercal \boldsymbol{AQ} = \boldsymbol{H}. \tag{9}$$

Again, in practice, it may be very difficult to compute this entire expression forcing us to search for approximative alternatives. Consider equation 9 for which the only first k columns of $\boldsymbol{Q}$ have been computed. Let $\boldsymbol{Q}_k = [\boldsymbol{q}_1, \boldsymbol{q}_2, \ldots, \boldsymbol{q}_k]$ and $\boldsymbol{Q}_u = [\boldsymbol{q}_{k+1}, \boldsymbol{q}_{k+2}, \ldots, \boldsymbol{q}_n]$. Then

$$\begin{aligned}\boldsymbol{Q}^\intercal \boldsymbol{AQ} &= \boldsymbol{H}\\ [\boldsymbol{Q}_k, \boldsymbol{Q}_u]^\intercal \boldsymbol{A} [\boldsymbol{Q}_k, \boldsymbol{Q}_u] &= \begin{bmatrix}\boldsymbol{H}_k & \boldsymbol{H}_{u,k}\\ \boldsymbol{H}_{k,u} & \boldsymbol{H}_u\end{bmatrix}\\ \begin{bmatrix}\boldsymbol{Q}_k^\intercal \boldsymbol{AQ}_k & \boldsymbol{Q}_k^\intercal \boldsymbol{AQ}_u\\ \boldsymbol{Q}_u^\intercal \boldsymbol{AQ}_k & \boldsymbol{Q}_u^\intercal \boldsymbol{AQ}_u\end{bmatrix} &= \begin{bmatrix}\boldsymbol{H}_k & \boldsymbol{H}_{u,k}\\ \boldsymbol{H}_{k,u} & \boldsymbol{H}_u\end{bmatrix}\end{aligned}$$

where $\boldsymbol{H}_k, \boldsymbol{H}_{u,k}, \boldsymbol{H}_{k,u}$ and $\boldsymbol{H}_u$ are the relevant sub matrices. This provides us with the equality

$$\boldsymbol{Q}_k^\intercal \boldsymbol{AQ}_k = \boldsymbol{H}_k \tag{10}$$

noting that $\boldsymbol{H}_k$ is upper Hessenberg for the same reason that $\boldsymbol{H}$ is. We know that when $n = t_{\boldsymbol{r}_0, \boldsymbol{A}}$ we can find a $\boldsymbol{Q} \in \mathbb{K}^{n \times n}$ and $\boldsymbol{H} \in \mathbb{K}^{n \times n}$ that satisfies $\boldsymbol{AQ} = \boldsymbol{QH}$. However, in general, we may not be so fortunate in finding a $\boldsymbol{Q}_k \in \mathbb{K}^{n \times k}$ and $\boldsymbol{H}_k \in \mathbb{K}^{n \times k}$ so satisfy $\boldsymbol{AQ}_k = \boldsymbol{Q}_k \boldsymbol{H}_k$ for any $k < n$. Instead we can adjust this equality by adding an error $\boldsymbol{E}_k \in \mathbb{K}^{n \times k}$ so that we do get equality. Our expression now becomes

$$\boldsymbol{Q}_k^\intercal \boldsymbol{AQ}_k = \boldsymbol{H}_k + \boldsymbol{E}_k. \tag{11}$$

A careful choice of $\boldsymbol{E}_k$ must be made to also retain equality in equation 10, meaning $\boldsymbol{Q}_k^\intercal \boldsymbol{E}_k = \boldsymbol{0}$. Since $\{\boldsymbol{q}_i\}_{i=1}^k$ forms an orthornormal basis for $\mathcal{K}_n(\boldsymbol{A}, \boldsymbol{r}_0)$, consider the following choice of $\boldsymbol{E}_k$,

$$\boldsymbol{E}_k = \boldsymbol{q}_{k+1} \boldsymbol{h}_k^\intercal$$

where $\boldsymbol{h}_k$ is any vector in $\mathbb{K}^k$. Notice that

$$\boldsymbol{Q}_k^\intercal \boldsymbol{E} = \boldsymbol{Q}^\intercal (\boldsymbol{q}_{k+1} \boldsymbol{h}_k) = (\boldsymbol{Q}^\intercal \boldsymbol{q}_{k+1}) \boldsymbol{h}_k^\intercal = \boldsymbol{0}.$$

Since this holds for any $\boldsymbol{h}_k \in \mathbb{K}^k$, to preserve sparsity and to keep this form as simple as possible we can set $\boldsymbol{h}_k = [0, 0, \ldots, h_{k+1,k}]^\intercal$. This means $\boldsymbol{AQ}_k$ can be written as

$$\boldsymbol{AQ}_k = \boldsymbol{Q}_k \boldsymbol{H}_k + \boldsymbol{q}_{k+1} \boldsymbol{h}_k^\intercal \tag{12}$$

where

$$\boldsymbol{Q}_k\boldsymbol{H}_k = [\boldsymbol{q}_1, \boldsymbol{q}_2, \ldots, \boldsymbol{q}_k] \begin{bmatrix} h_{1,1} & \cdots & \cdots & \cdots & h_{1,k} \\ h_{2,1} & \cdots & \cdots & \cdots & \vdots \\ 0 & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & h_{k,k-1} & h_{k,k} \\ 0 & \cdots & 0 & 0 & h_{k+1,k} \end{bmatrix}.$$

Equating the j^{th} columns of equation 12 yields

$$\boldsymbol{A}\boldsymbol{q}_j = \sum_{i=1}^{j+1} h_{i,j}\boldsymbol{q}_i.$$

Again since $\{\boldsymbol{q}_i\}_{i=1}^n$ form an orthornormal basis, multiplying both sides by $\boldsymbol{q}_m$ for $1 \leq m \leq j$ gives

$$\boldsymbol{q}_m^\intercal \boldsymbol{A}\boldsymbol{q}_j = \sum_{i=1}^{j+1} h_{i,j}\boldsymbol{q}_m^\intercal \boldsymbol{q}_i = h_{m,j}$$

and so

$$h_{j+1,j}\boldsymbol{q}_{j+1} = \boldsymbol{A}\boldsymbol{q}_j - \sum_{i=1}^{j} h_{i,j}\boldsymbol{q}_i. \tag{13}$$

From equation 13 we find that $\boldsymbol{q}_{j+1}$ can be computed using a recurrance involving its previous Krylov factors. Notice this bears a striking resemblance to equation 6 having a virtually an identical setup to computing an orthornormal basis using the modified Gram-Schmidt process (algorithm 2). As such, values for $\boldsymbol{q}_{j+1}$ and $h_{j+1,j}$ can be evaluted using a procedure very similar to the modified Gram-Schmidt process better known as the Arnoldi algorithm [Tre97, Dem97], presented in algorithm 3.

Algorithm 3: Arnoldi Algorithm

input : $\boldsymbol{A}, \boldsymbol{r}_0$ and k, the number of columns of $\boldsymbol{Q}$ to compute.
output: $\boldsymbol{Q}_k, \boldsymbol{H}_k$.

$\boldsymbol{q}_1 = \boldsymbol{r}_0 / \|\boldsymbol{r}_0\|$
for $j = 1$ **to** k **do**
  $\boldsymbol{z} = \boldsymbol{A}\boldsymbol{q}_j$
  for $i = 1$ **to** j **do**
    $h_{i,j} = \langle \boldsymbol{q}_i, \boldsymbol{z} \rangle$
    $\boldsymbol{z} = \boldsymbol{z} - h_{i,j}\boldsymbol{q}_i$
  end
  $h_{j+1,j} = \|\boldsymbol{z}\|$
  if $h_{j+1,j} = 0$ **then**
    return $\boldsymbol{Q}_k, \boldsymbol{H}_k$
  end
  $\boldsymbol{q}_{j+1} = \boldsymbol{z} / h_{j+1,j}$
end
return $\boldsymbol{Q}_k, \boldsymbol{H}_k$

When $\boldsymbol{A}$ is symmertic then $\boldsymbol{H} = \boldsymbol{T}$ becomes a tridiagonal matrix, simplifying a large amount of the Arnoldi algorithm since a considerably large number of matrix elements from $\boldsymbol{T}$ can be written as

$$\boldsymbol{T} = \begin{bmatrix} \alpha_1 & \beta_1 & & & \\ \beta_1 & \ddots & \ddots & & \\ & \ddots & \ddots & \ddots & \\ & & \ddots & \ddots & \beta_{n-1} \\ & & & \beta_{n-1} & \alpha_n \end{bmatrix}.$$

As before, equating the j^{th} columns of $\boldsymbol{AQ} = \boldsymbol{QT}$ yields

$$\boldsymbol{A}\boldsymbol{q}_j = \beta_{j-1}\boldsymbol{q}_{j-1} + \alpha_j \boldsymbol{q}_j + \beta_j \boldsymbol{q}_{j+1}. \tag{14}$$

Again since $\{\boldsymbol{q}_i\}_{i=1}^n$ form an orthornormal basis, multiplying both sides of equation 14 by $\boldsymbol{q}_j$ gives $\boldsymbol{q}_j = \boldsymbol{A}\boldsymbol{q}_j = \alpha_j$. A simplified version of the Arnoldi algorithm can be devised can be used to compute $\{\boldsymbol{q}_i\}_{i=1}^n$ and $\boldsymbol{T}$ for symmetric matrices known as the Lanczos algorithm [Dem97]. The Lanczos algorithm is presented in algorithm 4.

Algorithm 4: Lanczos Algorithm

input : $\boldsymbol{A}, \boldsymbol{r}_0$ and k, the number of columns of $\boldsymbol{Q}$ to compute.
output: $\boldsymbol{Q}_k, \boldsymbol{T}_k$.

$\boldsymbol{q}_1 = \boldsymbol{r}_0/\|\boldsymbol{r}_0\|$, $\beta_0 = 0$, $\boldsymbol{q}_0 = 0$
for $j = 1$ **to** k **do**
  $\boldsymbol{z} = \boldsymbol{A}\boldsymbol{q}_j$
  $\alpha_j = \langle \boldsymbol{q}_j, \boldsymbol{z} \rangle$
  $\boldsymbol{z} = \boldsymbol{z} - \alpha_j \boldsymbol{q}_j - \beta_{j-1}\boldsymbol{q}_{j-1}$
  $\beta_j = \|z\|$
  if $\beta_j = 0$ **then**
    return $\boldsymbol{Q}_k, \boldsymbol{T}_k$
  end
  $\boldsymbol{q}_{j+1} = \boldsymbol{z}/\beta_j$
end
return $\boldsymbol{Q}_k, \boldsymbol{T}_k$

2.4. **Optimality Conditions.** So far we have shown that $\boldsymbol{x}^\star \in \mathcal{K}_{t_{\boldsymbol{r}_0},\boldsymbol{A}}(\boldsymbol{A}, \boldsymbol{r}_0)$ where $n = t_{\boldsymbol{r}_0}$ is the grade of $\boldsymbol{r}_0$ with respect to $\boldsymbol{A}$. Moreover from section 2.3 we found ways to construct a basis for $\mathcal{K}_{t_{\boldsymbol{r}_0},\boldsymbol{A}}(\boldsymbol{A}, \boldsymbol{r}_0)$ allowing us to generate vectors with these affine spaces, namely the Arnoldi algorithm (algorithm 3)and Lanczos algorithm (algorithm 4) for non-symmertic and symmertic systems respectively. From now on $\mathcal{K}_{t_{\boldsymbol{r}_0},\boldsymbol{A}}(\boldsymbol{A}, \boldsymbol{r}_0)$ will be abbreviated to $\mathcal{K}_{t_{\boldsymbol{r}_0},\boldsymbol{A}}$ when the context is clear. The question still remains however, how should one choose an $\boldsymbol{x}_k$ that best approximates $\boldsymbol{x}^*$ satisfying equation 5? Here are a few of the most well known methods for selecting a suitable $\boldsymbol{x}_k$.

(1) Select an $\boldsymbol{x}_k \in \boldsymbol{x}_0 + \mathcal{K}_k$ which minimizes $\|\boldsymbol{x}_k - \boldsymbol{x}^*\|_2$. While this method seems like the most intuitive and natural way to select $\boldsymbol{x}_k$, it is unfortunately of no practical use since there is not enough information in the Krylov subspace to find an $\boldsymbol{x}_k$ which matches this specification.
(2) Select an $\boldsymbol{x}_k \in \boldsymbol{x}_0 + \mathcal{K}_k$ which minimizes $\|\boldsymbol{r}_k\|_2$ (recall is the residual of $\boldsymbol{x}_k$, that is, $\boldsymbol{r}_k = \boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}_k$). This method is possible to implement. Two well known algorithms stem from this class of methods, notably MINRES (minimum residual) and GMRES (general minimum residual) which solve linear systems for symmetric and non-symmertic $\boldsymbol{A}$ respectively.
(3) When $\boldsymbol{A}$ is a positive definite matrix it defines a norm $\|r\|_{\boldsymbol{A}} = (\boldsymbol{r}^\intercal \boldsymbol{A}\boldsymbol{r})^{\frac{1}{2}}$, called the energy norm. Select an $\boldsymbol{x}_k \in \boldsymbol{x}_0 + \mathcal{K}_k$ which minimizes $\|r\|_{\boldsymbol{A}^{-1}}$ which is equivalent to minimizing $\|\boldsymbol{x}_k - \boldsymbol{x}\|_A$. This technique is known as the CG (conjugate gradient) algorithm.
(4) Select an $\boldsymbol{x}_k \in \boldsymbol{x}_0 + \mathcal{K}_k$ for which $\boldsymbol{r}_k \perp \mathcal{W}_k$ where $\mathcal{W}_k$ is some k-dimensional subspace. Two well known algorithms that belong to this family of methods are SYMMLQ (Symmetric LQ Method) and a variant of GMRES used for solving symmetric and non-symmetric methods respectively.

Interestingly, when $\boldsymbol{A}$ is symmetric positive definite and $\mathcal{W}_k = \mathcal{K}_k$ the last two selection methods are equivalent. This is stated more precisely in thoerem

Theorem 21. *In the context of the above selection method, if* $\boldsymbol{A} \succ \mathbf{0}$ *and* $\mathcal{W}_k = \mathcal{K}_k$ *in method* (4) *then it produces the same* $\boldsymbol{x}_k$ *in method* (3) [Dem97].

In fact the very last method can be used to bring together a number of different analytical aspects and unify them in a general framework known as projection methods. Selecting an $\boldsymbol{x}_k$ from our Krylov subspace allows k degrees of freedom meaning k constraints must be used to determine a unique $\boldsymbol{x}_k$ for selection. As seen in method (4) already, typically orthogonality constraints are imposed on the residual $\boldsymbol{r}_k$. Specifically we would like to find a $\boldsymbol{x}_k \in \boldsymbol{x}_0 + \mathcal{K}_k$ where $\boldsymbol{r}_k \perp \mathcal{W}_k$. This is sometimes referred to as the Petrov-Galerkin (or just Galerkin) conditions. Projection methods for which $\mathcal{W}_k = \mathcal{K}_k$ are known as orthogonal projections while methods for which $\mathcal{W}_k = \boldsymbol{A}\mathcal{K}_k$ are known as oblique projections. If we set $\boldsymbol{x}_k = \boldsymbol{x}_0 + \boldsymbol{z}_k$ for some $\boldsymbol{z}_k \in \mathcal{K}_k$ then the Petrov-Galerkin conditions imply $\boldsymbol{r}_0 - \boldsymbol{A}\boldsymbol{z}_k \perp \mathcal{W}_k$, or alternatively $\langle \boldsymbol{r}_0 - \boldsymbol{A}\boldsymbol{z}_k, \boldsymbol{w} \rangle = 0$ for every $\boldsymbol{w} \in \mathcal{W}_k$. To impose these conditions it will help to have an appropriate basis for $\mathcal{K}$ and $\mathcal{W}$. Suppose we have access to such a basis where $\{\boldsymbol{q}_i\}_{i=1}^k$ and $\{\boldsymbol{w}_i\}_{i=1}^k$ are basis elements for $\mathcal{K}$ and $\mathcal{W}$ respectively. Let

$$\boldsymbol{K}_k \triangleq [\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_k] \in \mathbb{K}^{n \times k}$$
$$\boldsymbol{W}_k \triangleq [\boldsymbol{w}_1, \boldsymbol{w}_2, \ldots, \boldsymbol{w}_k] \in \mathbb{K}^{n \times k}$$

then the Petrov-Galerkin conditions can be imposed as follows

$$\boldsymbol{K}_k \boldsymbol{y}_k = \boldsymbol{z}_k, \quad \text{for some } \boldsymbol{y}_k \in \mathbb{K}^k$$
$$\boldsymbol{W}_k^\intercal \left(\boldsymbol{r}_0 - \boldsymbol{A}\boldsymbol{K}_k \boldsymbol{y}_k \right) = \mathbf{0}.$$

Moreover if $\boldsymbol{W}_k^\intercal \boldsymbol{A}\boldsymbol{K}_k$ is invertible then $\boldsymbol{x}_k$ can be expressed as

$$\boldsymbol{x}_k = \boldsymbol{x}_0 + \boldsymbol{K}_k \left(\boldsymbol{W}_k^\intercal \boldsymbol{A}\boldsymbol{K}_k \right)^{-1} \boldsymbol{W}_k \boldsymbol{r}_0. \tag{15}$$

This justifies a general form of the projection method algorithm presented in algorithm 5.

Algorithm 5: General Projection Method

output: An approximation of $\boldsymbol{x}^*$, $\boldsymbol{x}_k$.

for $k = 1, \ldots$ **until** *convergence* **do**
- Select $\mathcal{K}_k$ and $\mathcal{W}_k$
- Form $\boldsymbol{K}_k$ and $\boldsymbol{W}_k$
- Solve $\left(\boldsymbol{W}_k^\intercal \boldsymbol{A}\boldsymbol{K}_k \right) \boldsymbol{y}_k = \boldsymbol{W}_k^\intercal \boldsymbol{r}_0$
- $\boldsymbol{x}_k = \boldsymbol{x}_0 + \boldsymbol{K}_k \boldsymbol{y}_k$

end

return $\boldsymbol{x}_k$

2.5. **Conjugate Gradient Algorithm.** From section **??** that the Petrov-Galerkin conditions for the CG algorithm used an orthogonal projection and the matrix $\boldsymbol{A}$ was assumed to be positive definite. To derive the CG algorithm we can start be using some machinery that the Lanczos algorithm provides us with. Recall, the Lanczos algorithm produces the form $\boldsymbol{A}\boldsymbol{Q}_k = \boldsymbol{Q}_k \boldsymbol{T}_k + \boldsymbol{q}_{k+1} \boldsymbol{t}_k^\intercal$ where $\boldsymbol{t}_k \triangleq [0, 0, \ldots, 0, \beta_k]^\intercal \in \mathbb{K}^k$ and the columns of $\boldsymbol{Q}_k$ span $\mathcal{K}_k$. Recall that $\boldsymbol{x}_k$ can be expressed as $\boldsymbol{x}_k = \boldsymbol{x}_0 + \boldsymbol{K}_k \left(\boldsymbol{W}_k^\intercal \boldsymbol{A}\boldsymbol{K}_k \right)^{-1} \boldsymbol{W}_k \boldsymbol{r}_0$ (equation 15) when $\boldsymbol{W}_k^\intercal \boldsymbol{A}\boldsymbol{K}_k$ is invertible. For the CG algorithm $\mathcal{K} = \mathcal{W}$ and $\boldsymbol{A} \succ \mathbf{0}$. Under these

conditions we can easily show that $\boldsymbol{W}_k^\intercal \boldsymbol{A}\boldsymbol{K}_k$ is indeed invertible. This means the approximate vector can be expressed as $\boldsymbol{x}_k = \boldsymbol{x}_0 + \boldsymbol{z}_k$ where $\boldsymbol{z}_k \in \mathcal{K}_k$. In terms of the Petrov-Galerkin conditions this means that $\boldsymbol{z}_k$ must satisfy $\boldsymbol{r}_0 - \boldsymbol{A}\boldsymbol{z}_k \perp \mathcal{W}_k$. Furthermore since $\mathcal{K}_k = \text{Range}\left(\boldsymbol{Q}_k\right)$ where $\boldsymbol{Q}_k$ has full column rank then $\boldsymbol{z}_k$ can be represented as $\boldsymbol{z}_k = \boldsymbol{Q}_k\boldsymbol{y}$ for a unique $\boldsymbol{y} \in \mathbb{K}^k$ so that

$$\boldsymbol{x}_k = \boldsymbol{x}_0 + \boldsymbol{Q}_k\boldsymbol{y}. \tag{16}$$

Coupling this with the Petrov-Galerkin conditions means

$$\begin{aligned} \boldsymbol{Q}_k^\intercal\left(\boldsymbol{r}_0 - \boldsymbol{A}\boldsymbol{Q}_k\boldsymbol{y}\right) &= \boldsymbol{0} \\ \boldsymbol{Q}_k^\intercal\boldsymbol{A}\boldsymbol{Q}_k\boldsymbol{y} &= \boldsymbol{Q}_k^\intercal\boldsymbol{r}_0 \\ \boldsymbol{T}_k\boldsymbol{y} &= \|\boldsymbol{r}_0\|\boldsymbol{e}_1. \end{aligned} \tag{17}$$

In the CG algorithm $\boldsymbol{x}_{k+1}$ is computed as the recurrance of the following three sets of vectors

(1) The approximate solutions $\boldsymbol{x}_k$
(2) The residual vectors $\boldsymbol{r}_k$
(3) The conjugate gradient vectors $\boldsymbol{p}_k$

The conjugate gradient vectors are given the name gradient since the attempt to find the direction of steepest descent that minimizes $\|\boldsymbol{r}_k\|_{\boldsymbol{A}^{-1}}$. The are also given the name conjugate since $\langle \boldsymbol{p}_k, \boldsymbol{A}\boldsymbol{p}_j\rangle = 0$ for $i \neq j$, that is, vectors $\boldsymbol{p}_i$ and $\boldsymbol{p}_j$ are mutally A-conjugate.

Since $\boldsymbol{A}$ is symmetric positive definite then so is $\boldsymbol{T}_k = \boldsymbol{Q}_k\boldsymbol{A}\boldsymbol{Q}_k$. We can take the Cholesky decomposition of $\boldsymbol{T}_k$ to get

$$\boldsymbol{T}_k = \boldsymbol{L}_k\boldsymbol{D}_k\boldsymbol{L}_k^\intercal \tag{18}$$

where $\boldsymbol{L}_k$ is a unit lower bidiagonal matrix and $\boldsymbol{D}_k$ is diagonal written as

$$\boldsymbol{L}_k = \begin{bmatrix} 1 & & & \\ l_1 & \ddots & & \\ & \ddots & \ddots & \\ & & l_{k-1} & 1 \end{bmatrix}, \quad \boldsymbol{D}_k = \begin{bmatrix} d_1 & & & \\ & d_2 & & \\ & & \ddots & \\ & & & d_k \end{bmatrix}.$$

Combining equations 16, 17 and 18

$$\begin{aligned} \boldsymbol{x}_k &= \boldsymbol{x}_0 + \boldsymbol{Q}_k\boldsymbol{y} \\ \boldsymbol{x}_k &= \boldsymbol{x}_0 + \|\boldsymbol{r}_0\|\boldsymbol{Q}_k\boldsymbol{T}_k^{-1}\boldsymbol{e}_1 \\ \boldsymbol{x}_k &= \boldsymbol{x}_0 + \|\boldsymbol{r}_0\|\boldsymbol{Q}_k\left(\boldsymbol{L}_k\boldsymbol{D}_k\boldsymbol{L}_k^\intercal\right)^{-1}\boldsymbol{e}_1 \\ \boldsymbol{x}_k &= \boldsymbol{x}_0 + \left(\boldsymbol{Q}_k\boldsymbol{L}_k^{-\intercal}\right)\left(\|\boldsymbol{r}_0\|\boldsymbol{D}_k^{-1}\boldsymbol{L}_k^{-1}\boldsymbol{e}_1\right) \\ \boldsymbol{x}_k &\triangleq \boldsymbol{x}_0 + \tilde{\boldsymbol{P}}_k\tilde{\boldsymbol{y}}_k \end{aligned}$$

where $\tilde{\boldsymbol{P}}_k = \boldsymbol{Q}_k\boldsymbol{L}_k^{-\intercal}$ and $\tilde{\boldsymbol{y}}_k = \|\boldsymbol{r}_0\|\boldsymbol{D}_k^{-1}\boldsymbol{L}_k^{-1}\boldsymbol{e}_1$. The matrix $\tilde{\boldsymbol{P}}_k$ can be written as $\tilde{\boldsymbol{P}}_k = [\tilde{\boldsymbol{p}}_1, \tilde{\boldsymbol{p}}_2, \ldots, \tilde{\boldsymbol{p}}_k]$. Lemma 22 shows that the columns of $\tilde{\boldsymbol{P}}_k$ are A-conjugate.

Lemma 22. *The columns of $\tilde{\boldsymbol{P}}_k$ are A-conjugate, in otherwise $\tilde{\boldsymbol{P}}_k^\intercal\boldsymbol{A}\tilde{\boldsymbol{P}}_k$ is diagonal.*

Proof. We compute

$$\begin{aligned}
\tilde{\boldsymbol{P}}_k^\intercal \boldsymbol{A} \tilde{\boldsymbol{P}}_k &= \left(\boldsymbol{Q}_k \boldsymbol{L}_k^{-\intercal}\right)^\intercal \boldsymbol{A} \left(\boldsymbol{Q}_k \boldsymbol{L}_k^{-\intercal}\right) \\
&= \boldsymbol{L}_k^{-1} \left(\boldsymbol{Q}_k^\intercal \boldsymbol{A} \boldsymbol{Q}_k\right) \boldsymbol{L}_k^{-\intercal} \\
&= \boldsymbol{L}_k^{-1} \left(\boldsymbol{T}_k\right) \boldsymbol{L}_k^{-\intercal} \\
&= \boldsymbol{L}_k^{-1} \left(\boldsymbol{L}_k \boldsymbol{D}_k \boldsymbol{L}_k^\intercal\right) \boldsymbol{L}_k^{-\intercal} \qquad \text{(equation 18)} \\
&= \boldsymbol{D}_k
\end{aligned}$$

as wanted. □

Since $\boldsymbol{L}_k$ is a lower bidiagonal, setting $\boldsymbol{a} \triangleq l_{k-1}\boldsymbol{e}_{k-1}$, it can be written in the form

$$\boldsymbol{L}_k = \begin{bmatrix} \boldsymbol{L}_{k-1} & \boldsymbol{0} \\ \boldsymbol{a}^\intercal & 1 \end{bmatrix}$$

meaning

$$\boldsymbol{L}_k^{-1} = \begin{bmatrix} \boldsymbol{L}_{k-1}^{-1} & \boldsymbol{0} \\ \star & 1 \end{bmatrix}.$$

With this a recurrance for the columns of $\tilde{\boldsymbol{P}}_k$ can now be derived in terms of $\boldsymbol{y}_k$. To start we can show that the first $k-1$ entries of $\tilde{\boldsymbol{y}}_k$ shares the first $k-1$ entires with $\tilde{\boldsymbol{y}}_{k-1}$ and that $\tilde{\boldsymbol{P}}_k$ and $\tilde{\boldsymbol{P}}_{k-1}$ share the same first $k-1$ columns. To start we can compute a recurrance for $\tilde{\boldsymbol{y}}_k$ as follows

$$\begin{aligned}
\tilde{\boldsymbol{y}}_k &= \|\boldsymbol{r}_0\| \boldsymbol{D}_k^{-1} \boldsymbol{L}_k^{-1} \boldsymbol{e}_1^k \\
&= \|\boldsymbol{r}_0\| \begin{bmatrix} \boldsymbol{D}_{k-1}^{-1} & \boldsymbol{0} \\ \boldsymbol{0} & d_k^{-1} \end{bmatrix} \begin{bmatrix} \boldsymbol{L}_{k-1}^{-1} & \boldsymbol{0} \\ \star & 1 \end{bmatrix} \boldsymbol{e}_1^k \\
&= \|\boldsymbol{r}_0\| \begin{bmatrix} \boldsymbol{D}_{k-1}^{-1} \boldsymbol{L}_{k-1}^{-1} & \boldsymbol{0} \\ \star & d_k^{-1} \end{bmatrix} \begin{bmatrix} \boldsymbol{e}_1^k \\ 0 \end{bmatrix} \\
&= \begin{bmatrix} \tilde{\boldsymbol{y}}_{k-1} \\ \eta_k \end{bmatrix}
\end{aligned}$$

To get a recurrance for the columns of $\tilde{\boldsymbol{P}}_{k-1} = [\tilde{\boldsymbol{p}}_1, \tilde{\boldsymbol{p}}_2, \ldots, \tilde{\boldsymbol{p}}_k]$ since $\boldsymbol{L}_{k-1}^\intercal$ is upper triangular then so is $\boldsymbol{L}_{k-1}^{-\intercal}$, thus forming the leading $(k-1)-\text{by}-(k-1)$ submatrix of $\boldsymbol{L}_k^{-\intercal}$. This means that $\tilde{\boldsymbol{P}}_{k-1}$ is identical to the leading $k-1$ columns of

$$\tilde{\boldsymbol{P}}_k = \boldsymbol{Q}_k \boldsymbol{L}_k^{-\intercal} = [\boldsymbol{Q}_{k-1}, \boldsymbol{q}_k] \begin{bmatrix} \boldsymbol{L}_{k-1}^{-1} & \boldsymbol{0} \\ \star & 1 \end{bmatrix} = \left[\boldsymbol{Q}_{k-1}\boldsymbol{L}_{k-1}^{-1}, \tilde{\boldsymbol{p}}_k\right] = \left[\tilde{\boldsymbol{P}}_{k-1}, \tilde{\boldsymbol{p}}_k\right].$$

Moreover rearranging $\tilde{\boldsymbol{P}}_k = \boldsymbol{Q}_k \boldsymbol{L}_k^{-\intercal}$ we get $\tilde{\boldsymbol{P}}_k \boldsymbol{L}_k^\intercal = \boldsymbol{Q}_k$. Equating the k^{th} column yields

$$\tilde{\boldsymbol{p}}_k = \boldsymbol{q}_k - l_{k-1}\tilde{\boldsymbol{p}}_{k-1}. \tag{19}$$

Finally we can use

$$\boldsymbol{x}_k = \boldsymbol{x}_0 + \tilde{\boldsymbol{P}}_k \tilde{\boldsymbol{y}}_k = \boldsymbol{x}_0 + \left[\tilde{\boldsymbol{P}}_{k-1}, \tilde{\boldsymbol{p}}_k\right] \begin{bmatrix} \tilde{\boldsymbol{y}}_{k-1} \\ \eta_k \end{bmatrix} = \boldsymbol{x}_0 + \tilde{\boldsymbol{P}}_{k-1}\tilde{\boldsymbol{y}}_{k-1} + \eta_k \tilde{\boldsymbol{p}}_k = \boldsymbol{x}_{k-1} + \eta_k \tilde{\boldsymbol{p}}_k \tag{20}$$

as a recurrance for $\boldsymbol{x}_k$. A recurrance for $\boldsymbol{r}_k$ is easily computed as

$$\boldsymbol{r}_k = b - \boldsymbol{A}\boldsymbol{x}_k = b - \boldsymbol{A}\left(\boldsymbol{x}_{k-1} + \eta_k \tilde{\boldsymbol{p}}_k\right) = (b - \boldsymbol{A}\boldsymbol{x}_{k-1}) - \eta_k \boldsymbol{A}\tilde{\boldsymbol{p}}_k = \boldsymbol{r}_{k-1} - \eta_k \boldsymbol{A}\tilde{\boldsymbol{p}}_k \tag{21}$$

Altogether we are left with recurrences for $\boldsymbol{q}_k$ from Lanczos, $\tilde{\boldsymbol{p}}_k$ (equation 19), the residual $\boldsymbol{r}_k$ (equation 19), and for the approximate solution $\boldsymbol{x}_k$ (equation 20). However, futher simplification can be made for a more efficient algorithm. Recall from section 2.3 that $\boldsymbol{A}\boldsymbol{Q}_k = \boldsymbol{Q}_k\boldsymbol{T}_k + \boldsymbol{q}_{k+1}\boldsymbol{t}_k^\intercal$ where $\boldsymbol{t}_k = [0, 0, \ldots, 0, \beta_k]^\intercal \in \mathbb{K}^k$ meaning

$$\boldsymbol{r}_k = \boldsymbol{r}_0 - \boldsymbol{A}\boldsymbol{Q}_k\boldsymbol{y}_k = \boldsymbol{r}_0 - \boldsymbol{Q}_k\boldsymbol{T}_k\boldsymbol{y}_k - \langle \boldsymbol{t}_k, \boldsymbol{y}\rangle \boldsymbol{q}_{k+1} = -\beta_k y_k \boldsymbol{q}_{k+1}.$$

This tells us that $\boldsymbol{r}_k$ is parallel to $\boldsymbol{q}_{k+1}$ and orthogonal to all $\boldsymbol{q}_i,\ 1 \leq i \leq k$. This further implies that $\boldsymbol{r}_k$ is orthogonal to all $\boldsymbol{r}_i,\ 1 \leq i \leq k-1$ since they are just $\boldsymbol{q}_i$ scaled by some constant factor. So replacing $\boldsymbol{r}_{k-1}$ with $\boldsymbol{q}_k/\eta_k$ and defining $\boldsymbol{p}_k \triangleq \tilde{\boldsymbol{p}}_k/\gamma_k$ gives us a new set of recurrences

$$\begin{aligned}
\boldsymbol{x}_k &= \boldsymbol{x}_{k-1} + \alpha_k \boldsymbol{p}_k \\
\boldsymbol{r}_k &= \boldsymbol{r}_{k-1} - \alpha_k \boldsymbol{A}\boldsymbol{p}_k \\
\boldsymbol{p}_k &= \boldsymbol{r}_{k-1} + \beta_k \boldsymbol{p}_{k-1}
\end{aligned}$$

where $\alpha_k = \eta_k/\gamma_k$. From theorem 22 we have shown that the columns of $\tilde{\boldsymbol{P}}_k$ are A-conjugate (that is $\langle \tilde{\boldsymbol{p}}_i, \boldsymbol{A}\tilde{\boldsymbol{p}}_j\rangle = 0,\ i \neq j$) and that $\tilde{\boldsymbol{P}}_k^\intercal \boldsymbol{A}\tilde{\boldsymbol{P}}_k = \boldsymbol{D}_k$. This also means that $\langle \boldsymbol{r}_i, \boldsymbol{r}_j\rangle = 0,\ i \neq j$. Now note that from our recurrence for $\boldsymbol{p}_k = \boldsymbol{r}_{k-1} + \beta_k \boldsymbol{p}_{k-1}$ that

$$\langle \boldsymbol{A}\boldsymbol{p}_k, \boldsymbol{p}_k\rangle = \langle \boldsymbol{A}\boldsymbol{p}_k, \boldsymbol{r}_{k-1} + \beta_k \boldsymbol{p}_{k-1}\rangle = \langle \boldsymbol{A}\boldsymbol{p}_k, \boldsymbol{r}_{k-1}\rangle.$$

We can now find an expression for α_k as

$$\begin{aligned}
\langle \boldsymbol{r}_{k-1}, \boldsymbol{r}_k\rangle &= \langle \boldsymbol{r}_{k-1}, \boldsymbol{r}_{k-1} - \alpha_k \boldsymbol{A}\boldsymbol{p}_k\rangle \\
\langle \boldsymbol{r}_{k-1} - 1\rangle &= \langle \boldsymbol{r}_{k-1}, \boldsymbol{r}_{k-1}\rangle - \alpha_k \langle \boldsymbol{p}_k, \boldsymbol{A}\boldsymbol{p}_k\rangle \\
\alpha_k &= \frac{\langle \boldsymbol{r}_{k-1}, \boldsymbol{r}_{k-1}\rangle}{\langle \boldsymbol{p}_k, \boldsymbol{A}\boldsymbol{p}_k\rangle}.
\end{aligned}$$

Similarly, using the recurrence for $\boldsymbol{p}_k$, an expression for β_k can be computed as

$$\begin{aligned}
\langle \boldsymbol{A}\boldsymbol{p}_{k-1}, \boldsymbol{p}_k\rangle &= \langle \boldsymbol{A}\boldsymbol{p}_{k-1}, \boldsymbol{r}_{k-1} + \beta_k \boldsymbol{p}_{k-1}\rangle \\
\langle \boldsymbol{A}\boldsymbol{p}_{k-1}, \boldsymbol{p}_k\rangle &= \langle \boldsymbol{A}\boldsymbol{p}_{k-1}, \boldsymbol{r}_{k-1}\rangle + \beta_k \langle \boldsymbol{A}\boldsymbol{p}_{k-1}, \boldsymbol{p}_{k-1}\rangle \\
\beta_k &= -\frac{\langle \boldsymbol{A}\boldsymbol{p}_{k-1}, \boldsymbol{r}_{k-1}\rangle}{\langle \boldsymbol{A}\boldsymbol{p}_{k-1}, \boldsymbol{p}_{k-1}\rangle}
\end{aligned}$$

This formula requires am additional dot product which was not present before. Fortunately, this dot product can be eliminated using our recurrence for $\boldsymbol{r}_k$

$$\begin{aligned}
\langle \boldsymbol{r}_k, \boldsymbol{r}_k\rangle &= \langle \boldsymbol{r}_k, \boldsymbol{r}_{k-1} - \alpha_k \boldsymbol{A}\boldsymbol{p}_k\rangle \\
\langle \boldsymbol{r}_k, \boldsymbol{r}_k\rangle &= \langle \boldsymbol{r}_k, \boldsymbol{r}_{k-1}\rangle - \alpha_k \langle \boldsymbol{r}_k, \boldsymbol{A}\boldsymbol{p}_k\rangle \\
\alpha_k &= -\frac{\langle \boldsymbol{r}_k, \boldsymbol{r}_k\rangle}{\langle \boldsymbol{r}_k, \boldsymbol{A}\boldsymbol{p}_k\rangle}.
\end{aligned}$$

Equating the two expressions for $\boldsymbol{a}_k$ yields

$$-\frac{\langle \boldsymbol{r}_k, \boldsymbol{r}_k\rangle}{\langle \boldsymbol{r}_k, \boldsymbol{A}\boldsymbol{p}_k\rangle} = \frac{\langle \boldsymbol{r}_{k-1}, \boldsymbol{r}_{k-1}\rangle}{\langle \boldsymbol{p}_k, \boldsymbol{A}\boldsymbol{p}_k\rangle}$$

$$-\frac{\langle \boldsymbol{r}_k, \boldsymbol{r}_k \rangle}{\langle \boldsymbol{r}_{k-1}, \boldsymbol{r}_{k-1} \rangle} = \frac{\langle \boldsymbol{r}_k, \boldsymbol{A}\boldsymbol{p}_k \rangle}{\langle \boldsymbol{p}_k, \boldsymbol{A}\boldsymbol{p}_k \rangle}.$$

This means that

$$\beta_k = \frac{\langle \boldsymbol{r}_{k-1}, \boldsymbol{r}_{k-1} \rangle}{\langle \boldsymbol{r}_{k-2}, \boldsymbol{r}_{k-2} \rangle}.$$

These recurrences are computed iteratively to form the basis of the CG algorithm, seen in Algorithm 6.

Algorithm 6: CG Algorithm

input : $\boldsymbol{A} \succ \boldsymbol{0}$, $\boldsymbol{b}$ and an initial guess $\boldsymbol{x}_0$.

output: An approximation of $\boldsymbol{x}^*$, $\boldsymbol{x}_k$.

$\boldsymbol{r}_0 = \boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}_0$, $\boldsymbol{p}_1 = \boldsymbol{r}_0$

for $k = 1, \ldots$ **until** $\|r_{k-1}\| \leq \tau$ **do**

- $\alpha_k = \frac{\langle \boldsymbol{r}_{k-1}, \boldsymbol{r}_{k-1} \rangle}{\langle \boldsymbol{p}_k, \boldsymbol{A}\boldsymbol{p}_k \rangle}$
- $\boldsymbol{x}_k = \boldsymbol{x}_{k-1} + \alpha_k \boldsymbol{p}_k$
- $\boldsymbol{r}_k = \boldsymbol{r}_{k-1} - \alpha_k \boldsymbol{A}\boldsymbol{p}_k$
- $\beta_{k+1} = \frac{\langle \boldsymbol{r}_k, \boldsymbol{r}_k \rangle}{\langle \boldsymbol{r}_{k-1}, \boldsymbol{r}_{k-1} \rangle}$
- $\boldsymbol{p}_{k+1} = \boldsymbol{r}_k + \beta_{k+1} \boldsymbol{p}_k$

end

return $\boldsymbol{x}_k$

3. Twisted Hecke Algebras of Finite Groups

We have now completed our investigation of the Hecke algebra $\mathcal{H}(G,K)$. The aim of this section is to generalise the results of Chapter 1 to the case of a non-trivial character $\sigma\colon K \to \mathbb{C}^\times$. Here the Hecke algebra $\mathcal{H} = \mathcal{H}(G,K,\sigma)$ is the convolution algebra of (K,σ)-bi-invariant functions on G. In Section 3.1, we discuss the theory of the induced representation $\mathrm{Ind}_K^G \sigma$. In Section 3.2, we revisit the Hecke algebra, identify its identity and describe its basis. Notice that the results of Section 1.4 and Section 1.5 were independent of the choice $\sigma = \mathbf{1}$, so they still apply now that we are considering a non-trivial character.

In Section 3.3, we generalise Gelfand's Trick from Section **??** to the case of a non-trivial character σ. Naturally, we will need to reconsider the conditions that the anti-automorphism $\varphi\colon G \to G$ must satisfy. As in Section **??**, we will investigate these conditions and conclude with a natural statement and proof of Gelfand's Trick in the twisted case. We conclude with Section 3.4, in which we investigate the Gelfand–Graev representation and use the results of this chapter to prove that it is multiplicity-free.

3.1. The induced representation $\mathrm{Ind}_K^G \sigma$.

Suppose that $\sigma\colon K \to \mathbb{C}^\times$ is a character, i.e. a group homomorphism. Consider the space

$$W := \{f\colon G \to \mathbb{C} \mid f(gk) = f(g)\sigma(k),\ \forall g \in G, \forall k \in K\} \subseteq \mathrm{Fun}(G).$$

As in the previous section, W is called the induced representation and denoted $\mathrm{Ind}_K^G \sigma$. We state and prove a lemma analogous to Lemma **??**.

Lemma 23. *W is a left ideal of $(\mathrm{Fun}(G), \star)$.*

Proof. We verify that $f \star w \in W$ whenever $w \in W$ and $f \in \mathrm{Fun}(G)$. Let $g \in G$ and $k \in K$. Then

$$\begin{aligned}(f \star w)(gk) = \sum_{xy=gk} f(x)w(y) = \sum_{x\in G} f(x)w(x^{-1}gk) = \sum_{x\in G} f(x)w(x^{-1}g)\sigma(k)\\ = \left[\sum_{x\in G} f(x)w(x^{-1}g)\right]\sigma(k) = \left[\sum_{xy=g} f(x)w(y)\right]\sigma(k) = (f\star w)(g)\sigma(k). \quad \square\end{aligned}$$

3.2. The twisted Hecke algebra of a finite group $\mathcal{H}(G,K,\sigma)$.

The Hecke algebra $\mathcal{H} = \mathcal{H}(G,K,\sigma)$ is the space

$$\mathcal{H} := \{f\colon G \to \mathbb{C} \mid f(k_1gk_2) = \sigma(k_1)f(g)\sigma(k_2),\ \forall g \in G,\ \forall k_1,k_2 \in K\} \subseteq \mathrm{Fun}(G).$$

The proof of Lemma 23 can be adapted to show that $\mathcal{H}$ is a two-sided ideal in $(\mathrm{Fun}(G),\star)$. As before, the identity of $(\mathrm{Fun}(G),\star)$ does not lie in $\mathcal{H}$. Nevertheless, $\mathcal{H}$ does have an identity of its own. It is easy to verify that the identity is ι_K^σ, which we define below.

$$\iota_K^\sigma : G \to \mathbb{C}, \quad \iota_K^\sigma(g) := \begin{cases} \frac{1}{|K|}\sigma(g), & \text{if } g \in K,\\ 0, & \text{else.}\end{cases}$$

Thus, $(\mathcal{H},\star)$ is a unital associative algebra in its own right.

We now construct a basis for $\mathcal{H}$. Recall that when $\sigma = \mathbf{1}$, the basis of $\mathcal{H}$ was described by the characteristic functions of K-double cosets. To treat the case when $\sigma \neq \mathbf{1}$, we need a lemma about group actions.

Consider the finite group K acting on a set X. For each $x \in X$, let $\mathcal{O}_x := \{g \cdot x \mid g \in G\}$ be the orbit containing x and let $K_x := \{k \in K \mid k \cdot x = x\}$ be the stabiliser subgroup of x in K (also denoted as $\mathrm{stab}_K(x)$). Consider the vector space

$$V := \{f \colon X \to \mathbb{C} \mid f(k \cdot x) = \sigma(k) f(x),\ \forall k \in K,\ \forall x \in X\} \subseteq \mathrm{Fun}(X).$$

An orbit $\mathcal{O}_x$ is called (K, σ)-*relevant* if there exists $f \in V$ such that $f|_{\mathcal{O}_x}$ is non-zero. Otherwise, we say $\mathcal{O}_x$ is (K, σ)-*irrelevant*. We omit mention of (K, σ) if it is clear from the context.

Lemma 24. *An orbit $\mathcal{O}_x$ is (K, σ)-relevant if and only if $\sigma(K_x) = \{1\}$.*

Proof. Assume that $\sigma(K_x) \neq \{1\}$. Then there exists $k \in K_x$ such that $\sigma(k) \neq 1$. Now recall that for $f \in V$ we have $f(x) = f(k \cdot x) = \sigma(k) f(x)$. However $\sigma(k) \neq 1$, so $f(x) = 0$. Then f must be zero on $\mathcal{O}_x$ and $\mathcal{O}_x$ is irrelevant.

Conversely, the fact that σ is trivial on K_x implies that it factors through a well-defined function $\sigma_x \colon \mathcal{O}_x \simeq K/K_x \to \mathbb{C}$ given by $\sigma_x(kK_x) := \sigma(k)$. To see that this function is well-defined, suppose that $k_1 K_x = k_2 K_x$. Then $k_1 k_2^{-1} \in K_x$. Since σ is trivial on K_x, we know $1 = \sigma(k_1 k_2^{-1}) = \sigma(k_1)\sigma(k_2)^{-1}$. Then $\sigma(k_1) = \sigma(k_2)$ so $\sigma_x(k_1 K_x) = \sigma_x(k_2 K_x)$. It is easy to check that $\sigma_x \in V$. Thus, $\mathcal{O}_x$ is relevant. □

Now suppose K is a subgroup of G acting on $X = G$ from the left and right by translation. Then the orbit $\mathcal{O}_x$ is nothing but the double coset KxK and V becomes the Hecke algebra $\mathcal{H}$. Explicitly, we have

$$\mathcal{H} = \{f \colon X \to \mathbb{C} \mid f(k_1 \cdot x \cdot k_2) = \sigma(k_1) f(x) \sigma(k_2),\ \forall k_1, k_2 \in K,\ \forall x \in X\} \subseteq \mathrm{Fun}(X).$$

We can re-write this data by considering the left action of $K \times K^{\mathrm{op}}$ on X, where K^{op} is the group opposite to K. Then

$$\mathcal{H} = \{f \colon X \to \mathbb{C} \mid f((k_1, k_2) \cdot x) = \sigma(k_1)\sigma(k_2) f(x),\ \forall k_1, k_2 \in K,\ \forall x \in X\} \subseteq \mathrm{Fun}(X).$$

A double coset KxK is relevant if it supports a non-zero function from $\mathcal{H}$. Let X_{rel} be a family of relevant coset representatives. Define the family of functions $\{\chi_x\}_{x \in X_{\mathrm{rel}}}$ by

$$\chi_x(y) := \begin{cases} \sigma(k)\sigma(k'), & \text{if } y \in KxK \text{ with } y = kxk', \\ 0, & \text{if } y \notin KxK. \end{cases}$$

One easily checks that χ_x is well-defined. We call χ_x a *twisted characteristic function* associated to the relevant orbit KxK. When $\sigma = \mathbf{1}$, every orbit is relevant and $\sigma(k)\sigma(k') = 1$, so we obtain the original characteristic functions described in Section 1.3. Define the map

$$\sigma \boxtimes \sigma \colon K \times K \to \mathbb{C}^\times \times \mathbb{C}^\times, \quad (\sigma \boxtimes \sigma)(k_1, k_2) := (\sigma(k_1), \sigma(k_2)).$$

As a result of Lemma 24, we see that an orbit under the left action of $K \times K^{\mathrm{op}}$ is relevant if and only if $(\sigma \boxtimes \sigma)(\mathrm{stab}_{K \times K}(x)) = \{1\}$. As in Section 1.3, it is not difficult to see that the twisted characteristic functions of relevant orbits form a basis of $\mathcal{H}(G, K, \sigma)$.

3.3. **Twisted Gelfand's Trick.** Our goal in this section is to prove the twisted analogue of Gelfand's Trick.

Theorem 25 (Twisted Gelfand's Trick)**.** *Suppose that G is a finite group with $K \leq G$ as a subgroup and character $\sigma\colon K \to \mathbb{C}^\times$. Let $\varphi\colon G \to G$ be an anti-automorphism such that*

(i) $\varphi^2 = 1$,
(ii) $\varphi(K) = K$,
(iii) $\sigma(\varphi(k)) = \sigma(k)$ *for all* $k \in K$, *and*
(iv) $\varphi(x) = x$ *for all* $x \in X_{\mathrm{rel}}$, *a family of representatives for the* (K, σ)*-relevant* K*-double cosets.*

Then $\mathcal{H}(G, K, \sigma)$ is commutative.

This is a true generalisation of Theorem **??**. Indeed, if we consider the trivial representation $\sigma = \mathbf{1}$, condition *(iii)* is trivially satisfied, condition *(iv)* corresponds to the requirement that $K\varphi(x)K = KxK$ in Theorem **??**, and condition *(ii)* is contained in the requirement that $K\varphi(x)K = KxK$.

As in Section 3.1, the proof of Theorem 25 relies on the observation that an anti-homomorphism of an algebra that acts as the identity on basis elements of the subalgebra is sufficient to conclude that the subalgebra is commutative (c.f. Lemma **??** and Corollary **??**). This leaves us with a question: can we rewrite the condition $\varphi^*\chi_x = \chi_x$?

Recall that X_{rel} denotes a family of representatives for the relevant double cosets. Recall the twisted characteristic functions $\{\chi_x\}_{x \in X_{\mathrm{rel}}}$ defined in Section 3.2 given by

$$\chi_x(y) = \begin{cases} \sigma(k)\sigma(k'), & \text{if } y \in KxK \text{ with } y = kxk', \\ 0, & \text{if } y \notin KxK. \end{cases}$$

Thus,

$$(\varphi^*\chi_x)(g) = \begin{cases} \sigma(k)\sigma(k'), & \text{if } \varphi(g) \in KxK \text{ with } \varphi(g) = kxk', \\ 0, & \text{else.} \end{cases}$$

If $\varphi\colon G \to G$ is an involutive homomorphism, then $\varphi(g) = kxk'$ is equivalent to $g = \varphi(k')\varphi(x)\varphi(k)$. If we further suppose that $\varphi(x) = x$ for all $x \in X_{\mathrm{rel}}$ and $\varphi(K) = K$, then $g = \varphi(k')\varphi(x)\varphi(k)$ is equivalent to $g = \varphi(k')x\varphi(k)$. Thus,

$$(\varphi^*\chi_x)(g) = \begin{cases} \sigma(\varphi(k'))\sigma(\varphi(k)), & \text{if } g \in KxK \text{ with } g = \varphi(k')x\varphi(k), \\ 0, & \text{else.} \end{cases}$$

This tells us that $\varphi^*\chi_x$ is also supported (i.e. non-zero) on KxK. Now let's also assume that $\sigma(\varphi(k)) = \sigma(k)$ for all $k \in K$. Then we can easily verify that $\varphi^*\chi_x \in \mathcal{H}(G, K, \sigma)$. So $\varphi^*\chi_x$ must be a multiple of χ_x. In fact, this multiple is 1, since

$$(\varphi^*\chi_x)(x) = \chi_x(\varphi(x)) = \chi_x(x) = 1.$$

We are now ready to prove Theorem 25.

Proof of Theorem 25. Suppose that $\varphi\colon G \to G$ is an anti-automorphism. Also suppose that $\varphi^2 = 1$, $\varphi(K) = K$, $\sigma(\varphi(k)) = \sigma(k)$ for all $k \in K$, and $\varphi(x) = x$ for all $x \in X_{\mathrm{rel}}$. The above discussion tells us that

$\varphi^* \chi_x = \chi_x$. These are the basis elements of $\mathcal{H}(G, K, \sigma)$. We apply Corollary **??** to conclude that $\mathcal{H}(G, K, \sigma)$ is commutative. □

3.4. **The Gelfand–Graev representation.** We construct the *Gelfand–Graev representation* of $G = \mathrm{GL}_n(\mathbb{F}_q)$. First, consider the *unipotent radical* of G, given by

$$U(\mathbb{F}_q) := \begin{pmatrix} 1 & \mathbb{F}_q & \dots & \mathbb{F}_q \\ 0 & 1 & \ddots & \vdots \\ \vdots & \ddots & 1 & \mathbb{F}_q \\ 0 & \dots & 0 & 1 \end{pmatrix}.$$

Next, fix a non-trivial additive character $\psi \colon \mathbb{F}_q \to \mathbb{C}^\times$ (i.e. $\psi(a+b) = \psi(a)\psi(b)$). Then define a character $\pi \colon U(\mathbb{F}_q) \to \mathbb{C}^\times$ by

$$\pi(x) := \psi(x_{12} + x_{23} + \cdots + x_{n-1,n}).$$

To see that π is a character, observe

$$\begin{aligned}
\pi(xy) &= \psi((xy)_{12} + (xy)_{23} + \cdots + (xy)_{n-1,n}) \\
&= \psi\left(\sum_{k=1}^{n-1} x_{1k} y_{k2} + \sum_{k=1}^{n-1} x_{2k} y_{k3} + \cdots + \sum_{k=1}^{n-1} x_{n-1,k} y_{kn} \right) \\
&= \psi\left(\sum_{k=1}^{n-1} x_{1k} y_{k2} \right) \psi\left(\sum_{k=1}^{n-1} x_{2k} y_{k3} \right) \cdots \left(\sum_{k=1}^{n-1} x_{n-1,k} y_{kn} \right) \\
&= \psi(x_{12} + y_{12}) \psi(x_{23} + y_{23}) \cdots \psi(x_{n-1,n} + y_{n-1,n}) \\
&= \psi(x_{12}) \psi(y_{12}) \psi(x_{23}) \psi(y_{23}) \cdots \psi(x_{n-1,n}) \psi(y_{n-1,n}) \\
&= \psi(x_{12}) \psi(x_{23}) \cdots \psi(x_{n-1,n}) \psi(y_{12}) \psi(y_{23}) \cdots \psi(y_{n-1,n}) \\
&= \psi(x_{12} + \cdots + x_{n-1,n}) \psi(y_{12} + \cdots + y_{n-1,n}) \\
&= \pi(x) \pi(y).
\end{aligned}$$

The *Gelfand–Graev representation* of G is $\mathrm{Ind}_U^G \pi$. In [Bum13], Bump explains, "this Gelfand–Graev representation is important because it contains *most* irreducible representations of the group; those it contains are therefore called *generic*." Furthermore, we have the following theorem.

Theorem 26. *The Gelfand–Graev representation is multiplicity-free. That is, $(\mathrm{GL}_n(\mathbb{F}_q), U(\mathbb{F}_q), \pi)$ is a twisted Gelfand pair.*

This theorem will be proven in two parts. We begin with a lemma.

Lemma 27. (i) *We have the Bruhat decomposition*

$$\mathrm{GL}_n(\mathbb{F}_q) = \bigsqcup_{w \in W} BwB,$$

where W is the group of all $n \times n$ permutation matrices and B is the subgroup of all $n \times n$ upper-triangular matrices.

(*ii*) *We can modify the Bruhat decomposition and write*

$$\mathrm{GL}_n(\mathbb{F}_q) = \bigsqcup_{m \in M} UmU,$$

where M is the group of all $n \times n$ monomial matrices. A monomial matrix is a matrix with exactly one non-zero element in each row and column.

Before we prove Lemma 27, we recall a simple fact about matrices. Define $x_{ij}(t) := I_{n\times n} + tE_{ij}$, where $1 \leq i \leq j \leq n$ and E_{ij} is the matrix of 0's except for a 1 in the i^{th} row and j^{th} column. Notice that $x_{ij}(t) \in B$ since $i \leq j$. We can achieve the usual row and column operations on a matrix A by multiplying on the left or the right by some $x_{ij}(t)$. The following makes this statement precise.

Right-multiplying A by $x_{ij}(t)$ corresponds to the column operation of $C_j \mapsto C_j + tC_i$, where C_k is column k of A. Similarly, left-multiplying A by $x_{ij}(t)$ corresponds to the row operation of $R_i \mapsto R_i + tR_j$, where R_k is row k of A. Right-multiplying A by $x_{ii}(\lambda - 1)$ corresponds to the column operation $C_i \mapsto \lambda C_i$, for some scalar λ. Similarly, left-multiplying A by $x_{ii}(\lambda - 1)$ corresponds to the row operation $R_i \mapsto \lambda R_i$. We see that we can perform the usual row and column operations by right- and left-multiplying by elements of B.

Proof of Lemma 27. We begin by proving that $\mathrm{GL}_n(\mathbb{F}_q) = \bigcup_{w\in W} BwB$ and will prove disjointness of the union later. We proceed by induction. The $n = 1$ case is clearly true since all matrices in $\mathrm{GL}_1(\mathbb{F}_q)$ are upper-triangular. Now let $n > 1$ and $g \in \mathrm{GL}_n(\mathbb{F}_q)$. We wish to find a permutation matrix w in BgB. We have two cases: $g_{n,1} \neq 0$ and $g_{n,1} = 0$.

In the first case, the previous discussion tells us that we can multiply g on the left and the right by appropriate elements of B so that the resulting matrix has zeros in the left column and bottom row, except for the bottom left entry, which is $g_{n,1}$. This is non-zero so we can normalise this resulting matrix by $g_{n,1}$ to yield $\left(\begin{smallmatrix} 0 & g' \\ 1 & 0 \end{smallmatrix}\right)$. Here g' lies in $\mathrm{GL}_{n-1}(\mathbb{F}_q)$. The inductive hypothesis that the $n-1$ is true tells us that g' lies in a double coset $Bw'B$ for some $(n-1)\times(n-1)$ permutation matrix w'. Then the desired w is obtained by setting $w = \left(\begin{smallmatrix} 0 & w' \\ 1 & 0 \end{smallmatrix}\right)$.

In the second case, choose $g_{i1} \neq 0$ and $g_{nj} \neq 0$ so that i is as large as possible and j is as small as possible. This amounts to choosing the two non-zero entries in the left column and bottom row that are closest to the bottom left entry. Left- and right-multiplication by appropriate elements of B yields a matrix whose first and jth columns and ith and last rows are empty, except the entries g_{i1} and g_{nj}. Since these entries are non-zero, we can normalise these to 1 as well. Now we apply the inductive hypothesis to the matrix obtained by removing these two rows and two columns. We are left with a permutation matrix and this completes the induction.

We verify that the union is disjoint. Let $w_1, w_2 \in W$ be representatives for the same double coset. Then $Bw_1B = Bw_2B$ and, given any $b \in B$, there exists $b' \in B$ with $w_1 b w_2^{-1} = b'$. In particular, $w_1 w_2^{-1} \in B \cap W = \{1\}$. Thus $w_1 = w_2$.

We now prove the modified decomposition. Consider the subgroup T of diagonal matrices in $\mathrm{GL}_n(\mathbb{F}_q)$. Notice that $B = TU = UT$ and $M = TW = WT$ so the result follows from the regular Bruhat decomposition. Disjointness is proven as before. □

Proof of Theorem 26. Consider the involutive anti-automorphism $\varphi\colon G \to G$ defined by

$$\varphi(g) := w_0 g^t w_0, \quad \text{where } w_0 = \begin{pmatrix} & & 1 \\ & \iddots & \\ 1 & & \end{pmatrix}.$$

We verify that $U\varphi(g)U = UgU$ for all $g \in G$. For each double coset UgU, we will show that UgU has a certain coset representative g' with $\varphi(g') = g'$, or $f(g) = 0$ for all $f \in \mathcal{H}$.

The modification of the Bruhat decomposition in Lemma 27 tells us that $UgU = UmU$ for some monomial matrix m. Let $f \in \mathcal{H}$ be non-vanishing on UmU. That is, $f(m) \neq 0$. We show that m has the form

$$m = \begin{pmatrix} & & & D_1 \\ & & D_2 & \\ & \iddots & & \\ D_r & & & \end{pmatrix},$$

for some diagonal matrices $D_1,\ldots,D_r$. Equivalently, we show that if m_{ij} and $m_{i+1,k}$ are non-zero, then we must have $k \leq j+1$.

To see this, assume that $m_{ij}, m_{i+1,k} \neq 0$ and $k > j+1$. Then define $x := I_n + m_{ij}E_{i,i+1} \in U$ and $y := I_n + m_{i+1,k}E_{jk} \in U$. Simple computations tell us that $xm = m + m_{ij}m_{i+1,k}e_{ik} = my$, $\pi(x) = \psi(m_{ij}) \neq 1$ and $\pi(y) = \pi(0) = 1$. Then, since $f \in \mathcal{H}$, there holds $\pi(x)f(m) = f(xm) = f(my) = f(m)\pi(y)$. Thus $(\pi(x) - \pi(y))f(m) = 0$, so $f(m) = 0$ since $\pi(x) \neq \pi(y)$.

Now we show that each diagonal matrix D_i is actually a matrix of scalars. In particular, we show that if $m_{i,j}$ and $m_{i+1,j+1}$ are non-zero then they are equal. Consider x and y as given above, with $k = j+1$. Then $xm = my$, $\pi(x) = \psi(m_{ij})$, $\pi(y) = \psi(m_{i+1,j+1})$ and $(\pi(x) - \pi(y))f(m) = 0$. Recall that f doesn't vanish on UmU so $f(m) \neq 0$. Thus $\pi(x) = \pi(y)$, which tells us that $\psi(m_{ij}) = \psi(m_{i+1,j+1})$ and $m_{ij} = m_{i+1,j+1}$ by injectivity of ψ.

Finally, notice $\varphi(m) = m$. This is easy to see, since m^t is simply m with the elements on the opposite diagonal reversed, and left- and right-multiplying by w_0 also reverses the opposite diagonal. This completes the proof. $\square$

References

[Ras06] Carl Edward and Williams Rasmussen Christopher K. I, *Gaussian processes for machine learning / Carl Edward Rasmussen, Christopher K.I. Williams.*, Adaptive computation and machine learning, MIT Press, Cambridge, Mass., 2006 (eng).

[Mur12] Kevin P. Murphy, *Machine learning : a probabilistic perspective / Kevin P. Murphy.*, Adaptive computation and machine learning, MIT Press, Cambridge, MA, 2012 (eng).

[Ber96] Z.G. Sheftel Berezansky G.F, *Functional analysis. Volume 1 / Y.M. Berezansky, Z.G. Sheftel, G.F. Us ; translated from the Russian by Peter V. Malyshev.*, 1st ed. 1996., Operator Theory: Advances and Applications, 85, Basel ; Boston ; Berlin : BirkhaIuser Verlag, Basel ; Boston ; Berlin, 1996 (eng).

[Tre97] Lloyd N. (Lloyd Nicholas) and Bau Trefethen David, *Numerical linear algebra / Lloyd N. Trefethen, David Bau.*, SIAM Society for Industrial and Applied Mathematics, Philadelphia, 1997 (eng).

[Dem97] James W Demmel, *Applied numerical linear algebra / James W. Demmel.*, Society for Industrial and Applied Mathematics, Philadelphia, Pa., 1997 (eng).

[Ste08] Ingo and Christmann Steinwart Andreas, *Support Vector Machines*, 1st ed. 2008., Information Science and Statistics, Springer New York, New York, NY, 2008 (eng).

[Ber03] Alain and Thomas-Agnan Berlinet Christine, *Reproducing Kernel Hilbert Spaces in Probability and Statistics*, Springer, SpringerLink (Online service), Boston, MA, 2003 (eng).

[Ste99] Michael L Stein, *Interpolation of Spatial Data Some Theory for Kriging / by Michael L. Stein.*, 1st ed. 1999., Springer Series in Statistics, Springer New York : Imprint: Springer, New York, NY, 1999 (eng).

[Bos92] Bernhard and Guyon Boser Isabelle and Vapnik, *A training algorithm for optimal margin classifiers*, Proceedings of the fifth annual workshop on computational learning theory, 1992, pp. 144–152 (eng).

[Cor95] Corinna Cortes, *Support-Vector Networks*, Machine learning **20** (1995), no. 3, 273 (eng).

[AGS08] A. Aizenbud, D. Gourevitch, and E. Sayag, $(\mathrm{GL}_{n+1}(F), \mathrm{GL}_n(F))$ *is a Gelfand pair for any local field* F, Compositio Mathematica, Volume 144, Issue 6, 2008, pp. 1504–1524.

[BD85] T. Brocker and T. Dieck, *Representations of Compact Lie Groups*, Springer-Verlag, Berlin, 1985.

[BI84] E. Bannai and T. Ito, *Algebraic Combinatorics I : Association Schemes*, Benjamin Cummings Publishing Company, San Francisco, 1984.

[Bum10] D. Bump, *Hecke Algebras* (2010), http://sporadic.stanford.edu/bump/math263/hecke.pdf. Accessed 12/03/2021.

[Bum13] D. Bump, *Lie Groups*, Second Edition, Springer-Verlag, New York, 2013.

[Car85] R. W. Carter, *Finite Groups of Lie Type*, John Wiley & Sons Ltd, 1985.

[CSST20] T. Ceccherini-Silberstein, F. Scarabotti, and F. Tolli, *Gelfand Triples and Their Hecke Algebras: Harmonic Analysis for Multiplicity-Free Induced Representations of Finite Groups*, Lecture Notes in Mathematics, Springer International Publishing, 2020.

[CMHL03] I. Cherednik, Y. Markov, R. Howe, and G. Lusztig, *Iwahori-Hecke Algebras and Their Representation Theory*, Springer, Berlin, Heidelberg, 2003.

[Cox35] H. S. M. Coxeter, *The Complete Enumeration of Finite Groups of the Form* $R_i^2 = (R_i R_j)^{k_{ij}} = 1$ **s1-10** (1935), 21–25.

[CR87a] C. W. Curtis and I. Reiner, *Methods of Representation Theory - with applications to finite groups and orders*, Vol. 1, John Wiley & Sons, 1987.

[CR87b] C. W. Curtis and I. Reiner, *Methods of Representation Theory - with applications to finite groups and orders*, Vol. 2, John Wiley & Sons, 1987.

[Dia88] P. Diaconis, *Group representations in probability and statistics*, Institute of Mathematical Statistics, Hayward, California, 1988.

[EGH+11] P. I. Etingof, O. Golberg, S. Hensel, T. Liu, A. Schwendner, D. Vaintrob, and E. Yudovina, *Introduction to Representation Theory*, American Mathematical Society, 2011.

[Fol84] G. B. Folland, *Real Analysis*, John Wiley & Sons Inc., 1984.

[Gro91] B. H. Gross, *Some applications of Gelfand pairs to number theory*, Bulletin (New Series) of the American Mathematical Society **24** (1991), 277–301.

[Hen20] Y. I. Hendel, *On Twisted Gelfand Pairs Through Commutativity of a Hecke Algebra* (2020), https://arxiv.org/pdf/1807.02843.pdf. Accessed 20/01/21.

[HKP09] T. J. Haines, R. E. Kottwitz, and A. Prasad, *Iwahori–Hecke Algebras* (2009), https://arxiv.org/abs/math/0309168. Accessed 29/04/2021.

[HP02] T. J. Haines and A. Pettet, *Formulae relating the Bernstein and Iwahori–Matsumoto presentations of an affine Hecke algebra*, Journal of Algebra **252** (2002), 127–149.

[Hum90] J. E. Humphreys, *Reflection Groups and Coxeter Groups*, Cambridge University Press, New York, 1990.

[Kor80] A. Koranyi, *Some Applications of Gelfand Pairs in Classical Analysis*, Harmonic Analysis and Group Representation, 1980, pp. 334–348.

[Lam03] T. Y. Lam, *Corner Ring Theory: A Generalization of Peirce Decompositions, I*, Algebras, Rings and Their Representations: Proceedings of the International Conference on Algebras, Modules and Rings, 2003, pp. 153–182.

[Lan02] S. Lang, *Algebra*, revised 3rd edition, Springer-Verlag, New York, 2002.

[Mac95] I. G. Macdonald, *Symmetric Functions and Hall Polynomials*, 2nd ed., Oxford University Press Inc., New York, 1995.

[Mil20] J. S. Milne, *Algebraic Number Theory* (*v3.08*) (2020), www.jmilne.org/. Accessed 15/04/2021.

[Mor18] S. Morel, *MAT 449 : Representation theory* (2018), https://web.math.princeton.edu/~smorel/449/notes_449.pdf. Accessed 14/02/2021.

[MS21] Maarten Solleveld, *Affine Hecke Algebras and their Representations* (2021), https://arxiv.org/pdf/2009.03007.pdf. Accessed 31/05/21.

[Mur05] F. Murnaghan, *MAT445/1196F - Representation Theory Course Notes* (2005), http://www.math.toronto.edu/murnaghan/courses/mat445/notes.pdf. Accessed 15/02/2021.

[Mus93] C. Musili, *Representations of Finite Groups*, Texts and Readings in Mathematics, vol. 8, 1993, pp. 115–140.

[Pra05] A. Prasad, *On Bernstein's Presentation of Iwahori–Hecke Algebras and Representations of Split Reductive Groups over Non-Archimedian Local Fields* (2005), https://arxiv.org/pdf/math/0504417.pdf. Accessed 27/05/21.

[Rig96] L. J. Riggs, *Polynomial equations and solvability: A historical perspective*, California State University, San Bernardino, 1996.

[RW21] A. Romanov and G. Williamson, *Langlands correspondence and Bezrukavnikov's equivalence* (2021), https://arxiv.org/abs/2103.02329. Accessed 2/05/2021.

[Ter99] A. Terras, *Fourier Analysis on Finite Groups and Applications*, Cambridge University Press, London Mathematical Society, 1999.