Convexity is a Fundamental Feature of Efficient Semantic Compression in Probability Spaces.

Lindsay Skinner

A thesis
submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE

University of Washington

2025

Reading Committee:

Shane N Steinert-Threlkeld, Chair

Jakub Szymanik

Program Authorized to Offer Degree:
Linguistics

University of Washington

Abstract

Convexity is a Fundamental Feature of Efficient Semantic Compression in Probability Spaces.

Lindsay Skinner

Chair of the Supervisory Committee:
Assistant Professor Shane N Steinert-Threlkeld
Emily Bender

This thesis investigates the relationship between convexity and efficient communication using a probabilistic communication model applied to color space. It builds on previous work investigating the plausibility and potential source(s) of Gardenför's proposed semantic universal: that all subsets of color space affiliated with a particular color term are convex sets. The analysis undertaken in this project makes two major contributions to the existing literature.

- First, this project establish a new metric which defines a quantitative measure of convexity that can be applied to probabilistic communication models.

- Second, it demonstrates that convexity is an essential feature of efficient color-naming systems, where efficiency is determined with respect to a trade-off between accuracy and complexity. Furthermore, this project demonstrates that convexity is a more significant predictor of communication efficiency than either accuracy or complexity.

TABLE OF CONTENTS

Page

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGMENTS

The author wishes to express sincere appreciation to University of Washington, The CLMS program and Shane N Steinert-Threlkeld for his guidance throughout this project.

DEDICATION

To my partner Brandon, who encouraged and supported my pursuit of this degree from application to thesis.

Chapter 1

INTRODUCTION

Across the thousands of languages that exist in the world there is incredible variation in phonology, morphology, syntax and semantics. Despite these numerous differences, linguists have identified shared properties — linguistic universals — that occur in nearly every observed natural language. Studying these universals and the pressures that cause them to arise, while still allowing for significant linguistic variation, provides valuable insight into the connection between human language and cognition.

One such semantic universal is Gardenför's proposal that all natural properties are "convex" (Gardenfors, 2014). He argued that conceptual spaces can be understood geometrically and properties emerge as convex regions of these conceptual spaces (Gardenfors, 2000, 2014). This project focuses on the application of this hypothesis in conceptualizations of color; specifically the idea that the set of colors encapsulated by a single color term comprises a convex subset of color space (Gardenfors, 2000, 2014).

1.1 Convexity in Semantic Spaces

Many studies have explored Gardenför's theorem (Chemla et al., 2018; Gauker, 2007; Hernandez-Conde, 2016; Jäger, 2008, 2010; Steinert-Thelkeld and Szymanik, 2019) across different semantic categories, including quantifiers and color. The majority of this work has modeled conceptual spaces using discrete spaces, or continuous metric spaces divided into a discrete number of subspaces with hard boundaries. An example of this type of model is depicted in Figure 1.1. While these representations can be useful tools to explore the geometric properties of semantic universals, they do not fully account for the "messy" reality that one observes in real-world language use.

Semantic categories often have fuzzy boundaries, and disagreements between speakers of the same language are common. Look to color naming for an example; one can easily imagine

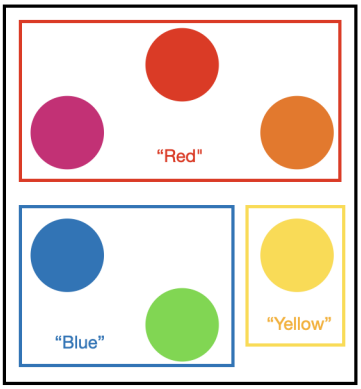


Figure 1.1: An example of a "typical" conceptual space representation with hard boundaries. In this simplified example there are six different color chips that define color space and three possible color terms that can be used to label each chip, "red", "blue", or "yellow". In this case, individual color terms define particular subsets of the set of color chips with hard boundaries between those subsets. Put differently, each chip is assigned to exactly one color term.

a scenario where two English speakers, who do not experience color vision deficiency, may readily agree when categorizing a particular color as "red" and another as "blue", but may disagree when faced with a cyan[1] observation and asked to categorize it as "blue" or "green". Furthermore, an individual may observe the same cyan color in different circumstances, e.g. surrounded by a collection of blue hues or surrounded by a collection of yellow hues, and categorize the color as "blue" under one set of circumstances but "green" under the other. These sorts of disagreements result in soft, uncertain boundaries that cause problems for geometric models of conceptual spaces which require clear boundaries between distinct properties.

Luckily, geometric spaces with hard boundaries are not the only available representations of conceptual spaces. Some studies have proposed probabilistic representations to represent semantic properties and explore linguistic universals (Zaslavsky et al., 2018). A toy example of one such representation is shown in Figure 1.2.

Probability spaces have the benefit of accounting for uncertainty or indecision among

[1]Cyan falls half-way between blue and green on the visual color spectrum. More information about the geometry of color spaces and this notion of "in-between-ness" will be provided in Section 3.4.

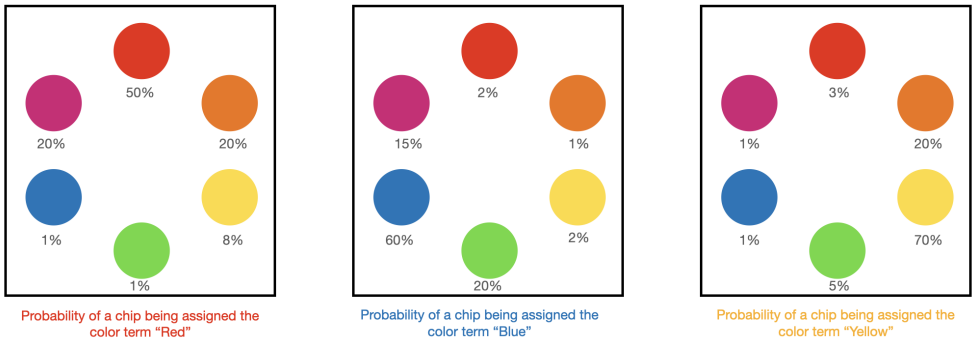


Figure 1.2: An example of a probabilistic representation of a conceptual space. As in Figure 1.1 there are six different color chips that define color space and three possible color terms that can be used to label each chip, "red", "blue", or "yellow". In this example individual color terms define probability distributions over the set of color chips. Each color term is more likely to refer to certain chips than others, but it is possible for multiple color terms to refer to the same chip in different contexts.

and between speakers. However, the notion of convexity over a probability space becomes more complicated than the definitions applied to Euclidean and discrete spaces.

1.2 Defining Convexity

The main quality defining convexity is the idea that if there are two points in a space that is convex with respect to a particular property, and both points have that property, then all points in between them will also share that property. This quality can manifest in numerous domains, three of which are described below:

Convex sets: all points on a line between two points in the set are also in the set. (Mokshay, 2011)

Convex functions: a function $f(x)$ is convex over X if and only if $f(tx_1 + (1-t)x_2) \leq tf(x_1) + (1-t)f(x_2), \forall x_1, x_2 \in X$. (Mokshay, 2011)

Convex measures: a convex measure μ does not assign more mass to any set C be-

tween[2] two sets A and B than it does to either A or B individually. (Mokshay, 2011; Laguel et al., 2021)

Since probabilistic spaces are metric spaces, the definition of a convex measure is particularly useful when exploring Gardenför's proposition in probabilistic representations of conceptual spaces. However, this definition simply defines whether or not a particular metric space is convex, it does not indicate "how" convex a given space may be. In order to determine the extent to which a particular metric space conforms to this definition of convexity, and thus be able to compare metric spaces in order to say that "space A is more or less convex than space B," we need a modified definition of convexity.

1.3 Major Contributions

This project adapts the definition of a convex measure in order to apply it to a particular probabilistic model of color semantics. This is accomplished using an optimal set of probabilistic encoders from an existing communication model, as well as a set of encoders derived from the natural languages in the World Color Survey (Cook et al.), in order to generate a wide-range of suboptimal encoders. We then apply an adapted definition of convexity in order to determine what, if any, role the convexity of an encoder plays in its optimality. This process results in two major contributions to the existing literature.

First Contribution: Establishes a new measure, the degree of quasi-convexity for probabilistic encoders, which defines a quantitative measure of convexity that can be applied to probabilistic models of semantics.

Second Contribution: Demonstrates that convexity is a key feature of communicative systems that are optimized for efficiency (i.e. the accuracy-complexity trade-off). Furthermore, the newly defined convexity measure has a greater impact than accuracy or complexity

[2]The notion of "betweenness" can be defined in a myriad of different ways, resulting in many different flavors of convexity. Here convex measures are defined in the most generic sense. Throughout the paper context-specific definitions of "betweenness" will be made explicit.

on the degree to which a particular system is optimally efficient.

Chapter 2

LITERATURE SURVEY

This project builds on a large collection of existing research on the topics of convexity, geometric representations of semantic spaces and the semantics of color. Key studies that influenced this project are summarized below; a complete list of influential work is included in the bibliography. The first section discusses papers that are relevant to outline the framework that is used to generate the color-naming encoders. The second section discusses papers that are influential in the development of the quasi-convexity measure that is used to quantify convexity in this project.

2.1 The Information Bottleneck Method

The Information-Bottleneck method was first described in (Tishby et al., 1999). This method outlines an algorithm for determining a short code (i.e. a bottleneck) for a signal variable that preserves the maximum information within the signal variable about a particular target variable.

The information bottleneck method was applied to explain the evolution of color-naming systems in natural languages in (Zaslavsky et al., 2018). This study shows that color-naming systems in natural languages "achieve near-optimally efficient compression, as predicted by the IB principle."(Zaslavsky et al., 2018, p.7937) Their findings extend beyond previous work to provide an explanation for the presence of soft categories and inconsistent naming that is seen in the empirical color-naming data.[1] The communication model outlined in this paper also demonstrates that optimal color naming systems undergo a sequence of phase transitions, which closely match patterns of color category evolution that have been observed among natural color-naming systems.(Zaslavsky et al., 2018, p.7941) Finally, (Zaslavsky et al., 2018) shows that small changes in the IB efficiency trade-off, between accuracy and

[1]The specific data source used is the World Color Survey data, which is discussed in detail in Section 3.1.

complexity, can account for much of the variation in color-naming systems across different natural languages.

The communication model from (Zaslavsky et al., 2018) is depicted in 2.1. In this case the signal variable is the speaker's conceptualization of a particular color, the distribution $m(u)$ over all points u in the color space, the short code is the color term used to convey that conceptualization, denoted by w, and the target variable is the listener's conceptualization, the distribution $\hat{m}(u)$ over all points u in the color space. The encoder, $q(w|m)$, is the speaker's strategy for assigning short codes, w, to meanings, $m(u)$. The decoder, $q(\hat{m}|w)$, is the listener's strategy for assigning meanings, $\hat{m}(u)$, to short codes, w. The IB-Optimal encoders are those that jointly maximize the accuracy of a color-naming system, defined by $I(U;W)$[2] and minimize the complexity, defined by $I(M;W)$[3].

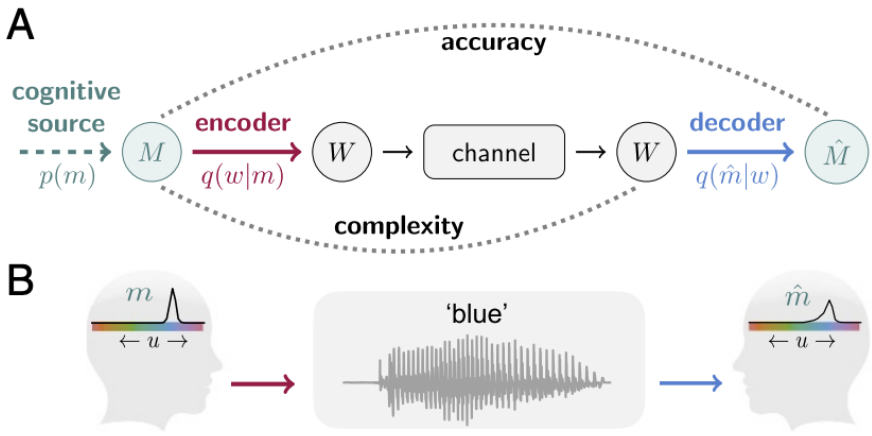


Figure 2.1: The IB model of communication as applied to color-space, taken from (Zaslavsky et al., 2018, p.7938)

The communication model, the IB function and the optimal encoders from (Zaslavsky et al., 2018) define optimal communication efficiency as it is used in this project. Further

[2]The mutual information of the colors and words; i.e. how much information the words contain about the target colors.

[3]The mutual information of the meanings and words, i.e. how much information the words contain about the speaker's conceptualization.

details about the communication model and information about the optimization algorithm and results can be found in (Zaslavsky et al., 2018) or at https://github.com/nogazs/ib-color-naming.

2.2 Degree of Convexity in Color Spaces

Color is a particularly useful domain for exploring convexity universals because it is highly structured and naturally maps to well-behaved geometric spaces. While many studies have explored the convexity of color space in relation to naturally occurring color naming systems (Chemla et al., 2018; Gardenfors, 2014), two studies that explored the degree of convexity of color-naming systems were particularly relevant to defining the convexity measure used in this project.

The first study of interest, (Jäger, 2010), uses the World Color Survey data[4] to create vector representations of color terms. Each vector is defined by the number of times a speaker assigns a color term to each of the 330 Munsell chips, resulting in a 330 dimensional vector for each term. Jäger then employs Principal Component Analysis (PCA) to reduce the dimension of the vector space and shows that categories of colors that map to the same color term in a given language are highly convex. For each color term he uses the PCA-reduced vectors to define convex sets of the CIELab color space that correspond to the specified term. He then introduces a definition for the "degree of convexity" which looks at how many color chips within a given convex set are correctly labeled by the term that defines that set.

The second study of interest, (Steinert-Thelkeld and Szymanik, 2019), uses neural networks to measure simplicity (i.e. ease of learning) in order to show that expressions satisfying semantic universals are simpler than those that do not satisfy semantic universals. In particular, they show this to be the case for two universals that apply to quantifiers (1. all simple determiners are quantitative; 2. all simple determiners are monotone) and one that applies to color naming systems (all color terms denote convex regions of color space). In their exploration of the convexity of color terms, (Steinert-Thelkeld and Szymanik, 2019)

[4]For a detailed explanation of the World Color Survey data see Section 3.1

defines the "degree of convexity" to measure how close each color-term-specified region is to its convex hull (i.e. what percentage of the convex hull is covered), aggregated across all the color terms in a system. They find that degree of convexity is the strongest predictor for learnability, "[there is] strong evidence that more convex color systems are indeed easier to learn."(Steinert-Thelkeld and Szymanik, 2019, p.7)

The degree of convexity definitions employed in both of these studies form the foundation for the degree of quasi-convexity measure that is used in this project, defined in Section 3.4. Additionally, the use of commonality analysis in Section 4.2 is inspired by similar analysis undertaken in (Steinert-Thelkeld and Szymanik, 2019).

Chapter 3

METHODOLOGY AND IMPLEMENTATION

This project explores the degree of convexity of natural language color-naming encoders, IB-optimal encoders, and sub-optimal encoders. In order to perform this analysis we generate encoders that represent the natural languages from available color-naming data and generated sub-optimal encoders from an available collection of IB-optimal encoders. We also defined a new metric to measure a probabilistic encoder's degree of quasi-convexity, in order to get a sense of "how convex" a particular encoder is compared to another. In order to facilitate this study the framework from (Piasini et al.) is implemented to reproduce the accuracy and complexity calculations defined in (Zaslavsky et al., 2018).[1] The details of these implementations are provided below.

3.1 Natural Language Encoders

The natural language encoders used in this project are derived from The World Color Survey data (Cook et al.). The World Color Survey is an exploration of color-naming systems across numerous languages. The survey consists of a set of color-categorization data collected across 110 different languages. For each language, an average of 24 native speakers were presented 330 Munsel chips of different colors and were asked to name the color of each chip. In addition to the color naming data, each chip includes a mapping to the affiliated CIELab coordinates. The data is freely available at https://linguistics.berkeley.edu/wcs/data.html.

The CIELab space is a subspace of $\mathbb{R}^3$ for which one dimension (L) represents lightness, the second dimension (a) represents the red-green axis and the third dimension (b) represents the yellow-blue axis. The space is standardized so that the Euclidean distances between colors correspond to the perceived dissimilarities between those colors by a human observer. The result is a geometric representation of color in a well-studied space ($\mathbb{R}^3$) that has a

[1] The full details of this implementation can be found at https://github.com/skinnel/color$_n aming_c onvexity$.

meaningful distance metric. This representation of color is used to explore the convexity of probabilistic encoders that represent natural and optimal color-naming strategies.

The natural language encoders are derived from this data. For each language in the dataset, counts are determined for each chip–color-term combination by looking at how many speakers assigned a specific color term to a specific chip. The counts are then normalized by color term, in order to generate a probability distribution across all of the 330 Munsell chips for each color term. The collection of these color term distributions forms the encoder for the specified language. Encoders are represented as matrices, where rows correspond to color chips, columns correspond to color terms and the cell value is the probability value determined by the color-term affiliated probability distribution.

3.2 IB-Optimal Encoders

The optimal encoders used in this paper are taken from (Zaslavsky et al., 2018). Each encoder consists of a set of color terms, each of which is associated with a probability distribution over the color space (defined by the 330 Munsel chips in the World Color Survey). The probability distribution for a specific color term indicates how likely a native speaker of the language represented by the encoder is to assign the specified color term to the color on the chip.

The optimality of an encoder is determined by the information-bottleneck trade off. "[A]n optimal encoder minimizes complexity by compressing the intended message M as much as possible, while maximizing the accuracy of its interpretation $\hat{M}$."(Zaslavsky et al., 2018, p.7938) Optimal encoders are derived via Rate Distortion Theory, the details of which can be found in (Tishby et al., 1999; Zaslavsky et al., 2018). This project uses the set of precomputed optimal encoders made freely available through https://github.com/nogazs/ib-color-naming.

3.3 Generating Sub-Optimal Encoders from the Optimal IB Encoders and Natural Languages

In order to investigate the relationship between efficient communication and convexity, this project requires encoders that vary in their optimality with respect to the IB trade-off

between accuracy and complexity. These sub-optimal encoders are generated by shuffling the data present in the Natural Language encoders and in the IB-Optimal encoders to varying degrees. These sub-optimal encoders enable a broader exploration of the accuracy-complexity space than the IB optimal and natural language encoders alone.[2]

Each encoder is represented as a matrix where rows correspond to chips and columns correspond to color terms. For each existing encoder, a certain number of rows in the encoder are randomly selected and shuffled in order to generate sub-optimal encoders. This process is depicted in Figure 3.1

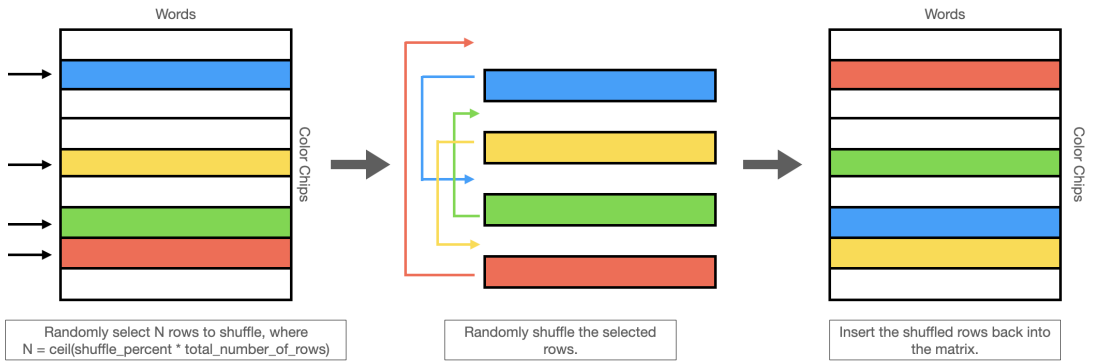


Figure 3.1: The process for randomly shuffling the rows of the optimal and natural language encoders in order to generate the sub-optimal encoders that fill out the complexity-accuracy space.

The benefit of this approach, as opposed to adding random noise to an encoder, is that shuffling the rows impacts the accuracy and complexity of an encoder while preserving some of the structure one would expect of a naturally-occurring color-naming system. These sub-optimal encoders are generated by randomly shuffling 5%, 10%, 15%, . . . 90%, 95%, 100% of the rows for each of the 1500 optimal encoders and 110 natural language encoders. For each shuffling percentage three random samples are used, resulting in 90,000 shuffled encoders

[2]The suboptimal encoders' complexity values range from 0.0004 to 7.2268, their accuracy values range from 0.014 to 4.283, and they range from lying on the IB-optimal frontier to being 1.724 units away from the frontier. The full breadth of the complexity-accuracy space that these encoders occupy is depicted in Figure 4.1

generated from the IB-optimal encoders and 6,600 shuffled encoders generated from the natural languages.

3.4 Quasi-Convexity

The main metric of interest in this project is the degree of quasi-convexity of a probabilistic encoder. This metric is designed to capture the same notion of convexity that has been applied in earlier studies — namely that if point a in color space is assigned "color term 1" and point b in color space is assigned "color term 1" than any point lying on the line between point a and point b is also assigned "color term 1" — but to do so for probabilistic encoders rather than sets with hard boundaries. This is accomplished by defining the act of "assigning a color chip a color term" in probabilistic terms. Specifically, the encoder for color term 1, P_1, is "convex" if and only if $\forall a, b, c \in X$, where X denotes color space and c is between[3] a and b, either $P_1(a) \leq P_1(c)$ or $P_1(b) \leq P_1(c)$. The effect of this definition is that if one is most likely to label the point a "color term 1" and most likely to label the point b "color term 1" then all points that fall between a and b in the CIELab color space are also most likely to be labeled "color term 1". In mathematical terms, a probability distribution that has this property is a quasi-concave[4] distribution (Mokshay, 2011).

We adapt this definition because it is a natural extension of the set-theoretical notion of convexity that appears in the hard-boundary conceptualizations of color space that were mentioned in Section 1.1. In fact, we can define probability distributions for the hard-boundary cases in such a way that the set of color naming strategies whose affiliated probability distributions are quasi-concave are exactly the set of color naming strategies that are convex. In those cases we define the probability distribution for a given color term as follows:

---

[3]Here, since the colors are defined over a subset of $\mathbb{R}^3$, "between" means $\exists t \in [0, 1]$ such that $c = t * a + (1 - t) * b$.

[4]The terminology gets a bit muddled here. The mathematical property we're interested in is "quasi-concavity," not "quasi-convexity". However, in order to maintain consistency with earlier work on this subject and emphasize that we're interested in the convexity of the level sets of the probability distribution, we adapt the expression "Degree of Quasi-Convexity" to describe our metric.

$$P_1(x) = \begin{cases} \frac{1}{n}, & \text{if color chip } x \text{ is assigned ``color term 1''} \\ 0, & \text{otherwise} \end{cases}$$

where n is the number of color chips assigned "color term 1".

If this color-naming system is quasi-concave then the effect is that if chip a is assigned "color term 1" and chip b is assigned "color term 1" then all chips between a and b will also be assigned "color term 1", which is equivalent to saying that the chips assigned "color term 1" define a convex set.

The encoders in this project are typically comprised of many different probability distributions, since each color term in a language invokes a separate distribution over the whole color space. This project is not interested in a binary classification of these individual distributions, but rather in a gradated metric that indicates the degree to which each encoder satisfies the quasi-concavity property. In order to generate this metric we first adapt the above definition to determine the degree to which each individual probability distribution satisfies the quasi-concavity property. Algorithm 1 shows the pseudo-code used to generate the quasi-convexity measure of a single probability distribution.

Algorithm 1 Calculate a Probability Distribution's Degree of Quasi-Convexity

Require: $mesh > 0$

Ensure: $P(x) \geq 0 \; \forall x \in X$

Ensure: $\sum_{x \in X} P(x) = 1.0$

$p \leftarrow 1.0$

$n \leftarrow \lceil \frac{p}{mesh} \rceil$

$qc \leftarrow 0$

for $i \in 0, \ldots, n$ **do**

$X_{level} \leftarrow \{x \mid P(x) \geq p\}$

$X_{hull} \leftarrow ConvHull(X_{level})$ ▷ Gets the convex hull of the points in X_{level}

$qc_{level} \leftarrow \frac{|X_{level}|}{|X_{hull}|}$ ▷ If X_{level} is empty we set $qc_{level} = 1.0$.

$qc \leftarrow qc + (mesh * qc_{level})$

$p \leftarrow p - mesh$

This adaptation extrapolates the "degree of convexity" definitions in (Jäger, 2010; Steinert-Thelkeld and Szymanik, 2019) to apply to probability distributions.

The encoder's degree of quasi-convexity is then determined using a weighted sum that is aggregated across all the color term distributions. The weights for each distribution are proportional to the number of color chips that assign maximal probability to the affiliated color term[5]. The resulting metric is a value between 0 and 1, where 1 indicates that every probability distribution that comprises the encoder is quasi-concave. The details of the aggregation can be found at https://github.com/skinnel/color$_n$*aming*$_c$*onvexity.Anexamplethatappliesthisalgorithmto*

[5]We use this weighting because it ensures that the impact to the degree of quasi-convexity calculation that comes from the probability distribution affiliated with a particular color term is proportional to the amount of color space that color term "covers." This way color terms that are affiliated with a larger subset of the color chips have a greater contribution to the final convexity measure than color terms that only apply to a few chips

Chapter 4

RESULTS

Now that we have everything in place we can analyze the connection between quasi-convexity and IB optimality through several lenses.

4.1 Analysis 1: Quasi-Convexity and IB Optimality

Looking across the complete set of natural language, optimal and sub-optimal encoders, one observes a high Pearson's correlation (0.754) between an encoder's degree of quasi-convexity and its optimality (negative distance from the IB frontier). Overall, encoders that are closer to the IB frontier have a higher degree of quasi-convexity, as depicted in Figure 4.1. This shows that encoders that are more optimal are more convex, suggesting that convexity in conceptual spaces is a fundamental property of efficient communication systems. We now turn to further analyses to disentangle other possible factor that could be contributing to this effect.

4.2 Analysis 2: Accuracy, Complexity and Quasi-Convexity vs. IB Optimality

When comparing frontier distance, degree of quasi-convexity (QC), accuracy and complexity it becomes apparent that all four variables are highly correlated. Table 4.1 shows that quasi-convexity is more highly correlated with optimality than accuracy and complexity, but it is also significantly correlated with accuracy. Because these four variables are correlated, two additional tests are used to determine how quasi-convexity, accuracy and complexity correlate with the frontier distance in the presence of the other variables.

The first test is a regression analysis, to get a sense of how correlated degree of quasi-convexity, accuracy and complexity are with frontier distance in the presence of the other variables. We fit a linear model to these three variables, with frontier distance as the target variable, using least-squares regression as the fitting criteria. The results are shown in

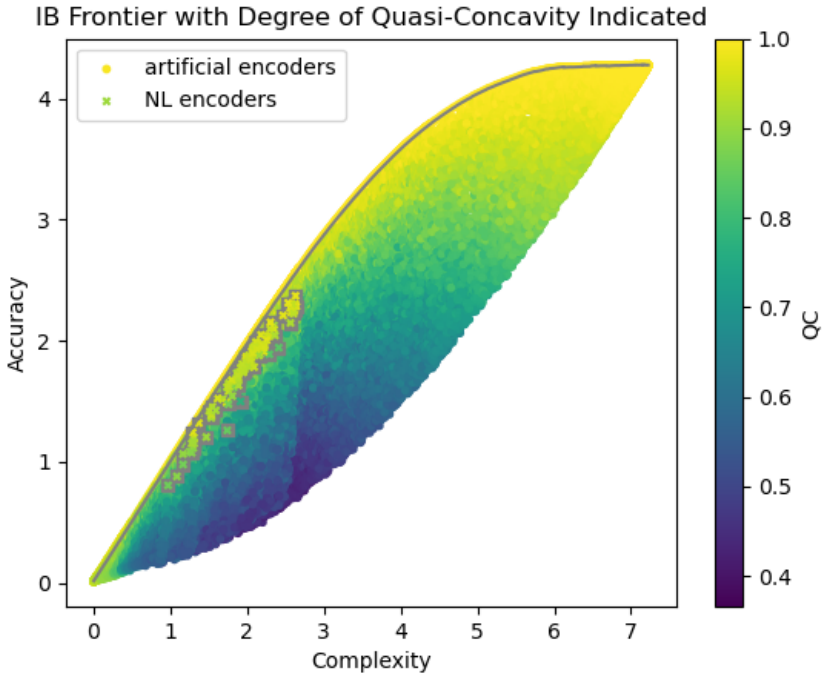


Figure 4.1: The IB frontier displaying the accuracy and complexity values for IB-optimal encoders, sub-optimal shuffled encoders, and the natural languages. The frontier is indicated by a gray line, synthetic decoders appear as dots and the natural languages are denoted by X marks with a gray border. The degree of quasi-convexity is denoted by color, with values closer to 1.0 (yellow) indicating a "more convex" encoder and values closer to 0.0 (purple) indicating a "less convex" encoder.

Correlation Values				
	Frontier Distance	QC	Accuracy	Complexity
Frontier Distance	1.0	-0.754	-0.456	-0.259
QC	-0.754	1.0	0.803	0.653
Accuracy	-0.456	0.803	1.0	0.954
Complexity	-0.259	0.653	0.954	1.0

Table 4.1: Table showing the correlations between frontier distance, degree of quasi-convexity, accuracy and complexity among the complete set of optimal, natural language and sub-optimal encoders. Note that optimality is defined to be the negative frontier distance, so negating the frontier distances correlations will yield the correlations with optimality.

Table 4.2. In this regression we see that all three variables and the constants term are statistically significant in the prediction of the frontier distance. Of those, the degree of quasi-convexity has the largest absolute value coefficient. In order to get a better sense of the individual impact each variable has on this prediction, we turn next to commonality analysis.

Commonality analysis is used to determine the unique impact that each variable has on the linear fit to frontier distance, beyond the predictive power of the other variables. This involves determining the R-squared value of a linear fit to the frontier distance that takes the degree of quasi-convexity, accuracy and complexity as independent variables, and comparing it to the R-squared value of a linear fit to the frontier distance that excludes the variable of interest. This difference gives an indication of the amount of variance in the frontier distance that can be explained by a single variable, beyond the variance that can be explained by the remaining variables.

The degree of quasi-convexity has a difference in R-squared values that is an order of magnitude higher than the differences in values for accuracy and complexity. This means that the unique effect of the degree of quasi-convexity in relation to the frontier distance, and thus optimality, is more significant than that of accuracy or complexity. Additional testing

Regression Values				
Variable	Coefficient	Std. Er.	t	p
Constant	1.6697	0.005	313.318	0.000
QC	-1.7793	0.009	-199.515	0.000
Accuracy	-0.2499	0.003	-76.384	0.000
Complexity	0.2090	0.002	124.818	0.000

Table 4.2: Table showing the regression coefficients, standard error, and statistical significance (t and p values) for the constant term, quasi-convexity (QC), accuracy and complexity in predicting frontier distance.

ΔR2 Values			
QC	Accuracy	Complexity	Accuracy & Complexity
0.133	0.019	0.052	0.115

Table 4.3: The difference between the R-squared value of a linear fit of all three variables against the frontier distances and the R-squared value of a linear fit that includes all but the variable(s) indicated in the column against the frontier distance.

reveals that the degree of quasi-convexity has a higher difference in R-squared values than the difference obtained when both accuracy and complexity are removed from the fit. This indicates that the degree of quasi-convexity has a stronger impact when predicting frontier distance than the collective impact of accuracy and complexity.

4.3 Analysis 3: Natural Languages

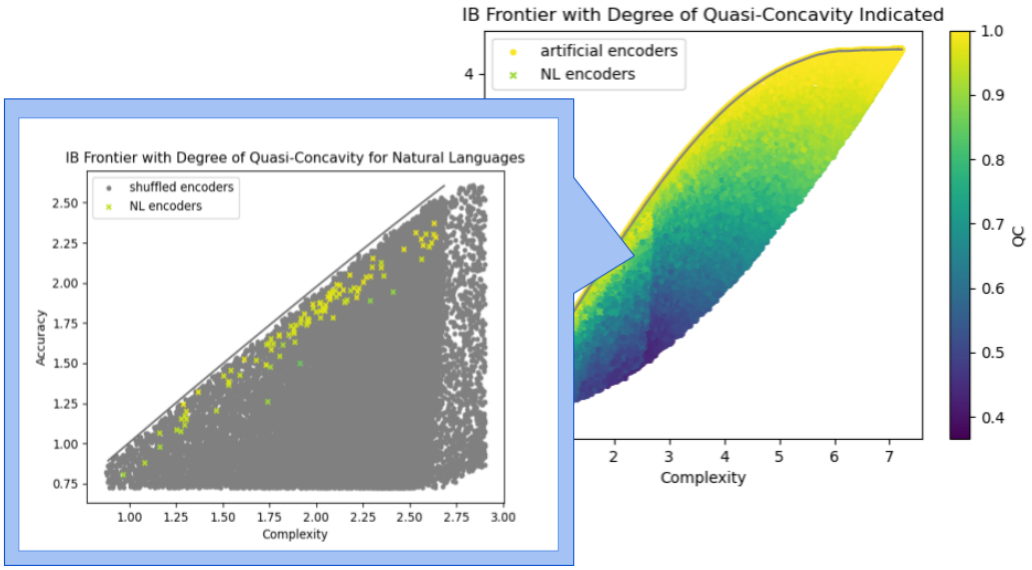


Figure 4.2: The IB frontier displaying the accuracy and complexity values for the encoders. The fronted plot is focused on the region that houses the natural languages and only shows the degree of quasi-convexity values for the natural languages, which are denoted by colored x marks. The synthetic sub-optimal encoders are marked by gray dots.

We now turn to a comparison of the encoder types in order to understand where the color-naming systems that arise from natural languages fall within this analysis.

Previous work in (Zaslavsky et al., 2018) shows that the natural languages are near-optimal when it comes to the IB tradeoff, which is apparent in Figure 4.2. It is also apparent in Table 4.4 and Figure 4.3, that the encoders derived from natural languages tend to be a higher-degree of quasi-convexity than sub-optimal encoders, all of which aligns with earlier analyses.

Frontier Distance, Accuracy, Complexity and Quasi-Convexity by Encoder Type					
Encoder Type	Frontier Distance Range	Accuracy Range	Complexity Range	QC Range	Average QC
Natural Languages	0.035 to 0.338	0.80 to 2.37	0.97 to 2.64	0.77 to 0.98	0.93
IB Optimal Encoders	0.000 for all	0.02 to 4.28	0.0 to 7.22	0.98 to 1.0	0.99
Shuffled Encoders	0.000 to 1.734	0.01 to 4.28	0.0 to 7.23	0.40 to 1.0	0.73

Table 4.4: The range of values for accuracy, complexity and degree of quasi-convexity, differentiated by encoder type.

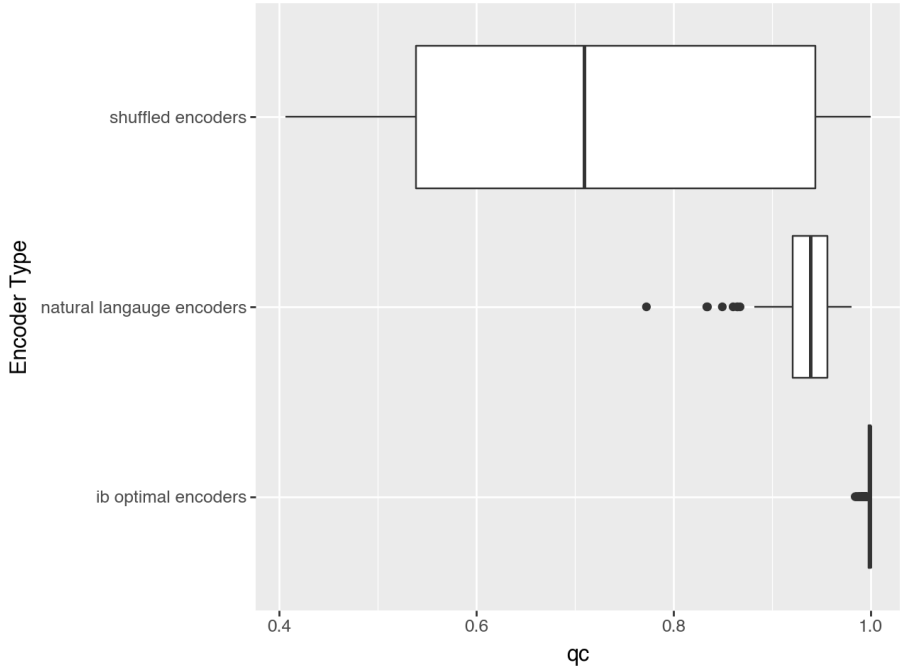


Figure 4.3: Box plots displaying the distributions of the degree of quasi-convexity for each type of encoder.

In comparison, the IB-optimal encoders are almost perfectly convex; their degree of quasi-convexity values range from 0.98 to 1.00, meeting or exceeding the degree of quasi-convexity of the natural language encoders in all cases. This is likely due to the fact that the natural language encoders are also less optimal in that they are farther from the IB-optimal frontier, as shown in Table 4.4. The less-than-perfect degree of quasi-convexity values for the natural language encoders can be explained by their deviations from optimality.

Chapter 5

DISCUSSION

The results indicate that an encoder's degree of quasi-convexity is not merely correlated with IB optimality, it explains the variance in optimality beyond the impact of accuracy and complexity. This means that if accuracy and complexity are held constant, more convex encoders are on average more optimal.

The results also demonstrate that encoders derived from the natural languages are closer to the optimal IB frontier and have a higher degree of quasi-convexity than the majority of the sub-optimal encoders[1]. The high correlation between the degree of quasi-convexity and optimality indicates that pressures for efficient communication shape natural languages to develop color-naming systems that are more convex. IB-optimal encoders are almost perfectly convex, and the more optimal an encoder is the more convex it is. This provides a plausible explanation for why so many languages manifest the semantic universal that color terms denote convex regions of color space.

Finally, this analysis is performed in a probabilistic space. This approach allows for an exploration of convexity using a representation that more closely matches the empirical data; it allows for fuzzy boundaries and inconsistencies among and between speakers. The relationship between communication efficiency and convexity described above emerges in a space that preserves the "fuzzy" qualities of the natural language data that many previous studies of convexity in color spaces do not account for. This analysis shows that communication efficiency pressures on natural languages provide a plausible explanation for Gardenför's proposed universal that fully accounts for the messiness and inconsistencies that are present in natural languages.

---

[1] 62.3% of the sub-optimal encoders are further from the IB frontier than all the natural language encoders and 53.9% of the sub-optimal encoders have a lower degree of quasi-convexity than all the natural language encoders.

Chapter 6
CONCLUSION AND FUTURE WORK

6.1 Conclusion

This project defines a new convexity measure that can be applied to probabilistic communication models and shows that convexity is an innate feature of efficient communication systems. In particular, the results show that color-naming systems that have a higher degree of quasi-convexity will be closer to the IB-optimal frontier than those that have a lesser degree of quasi-convexity. Furthermore, an encoder's degree of quasi-convexity has greater predictive power in determining its optimality than that encoder's accuracy, complexity, or both. All this leads us to conclude that convexity is an essential feature of optimality; the more convex a conceptual space is the more optimal, in terms of communication efficiency, it will be. Finally, this analysis shows that color-naming systems derived from the natural languages follow this pattern. Natural language encoders, which have previously been shown to be nearly IB-optimal, are far more convex than their sub-optimal counterparts but less convex than the optimal encoders. "Failures" in convexity among the natural language encoders can be attributed to deviations from optimality.

6.2 Future Work

Next steps in this project involve developing an end-to-end implementation of the IB optimization process that can derive the IB-optimal frontier from a random initialization, including all the encoders generated in intermediate steps. The goal of this implementation is to explore how the degree of quasi-convexity changes throughout the iterative process.

Another possible area for future work is demonstrating, via mathematical proof, under what conditions the optimal encoders derived from the iterative IB process will be quasi-convex. Additional work could also involve adapting the metric defined in this paper to apply to other semantic domains, e.g. person systems, quantifiers, etc.

BIBLIOGRAPHY

E. Chemla, B. Buccola, and I. Dautriche. Connecting content and logical words. *Journal of Semantics*, 36:531–547, August 2018.

R. Cook, P. Kay, and T. Regier. World color survey data archives. URL `https://linguistics.berkeley.edu/wcs/data.html`.

P. Gardenfors. *Conceptual Spaces: The Geometry of Thought*. The MIT Press, 2000.

P. Gardenfors. *The Geometry of Meaning*. The MIT Press, 2014.

C. Gauker. A critique of the similarity space theory of concepts. *Mind Language*, pages 317–345, 2007.

J. V. Hernandez-Conde. A case against convexity in conceptual spaces. *Synthese*, 194: 4011–4037, May 2016.

G. Jäger. The evolution of conovex categories. *Linguist and Philos*, 30:551–564, March 2008.

G. Jäger. Natural color categories are convex sets. In M. Aloni, H. Bastiaanse, T. de Jager, and K. Schulz, editors, *Logic, Language and Meaning*, pages 11–20, Berlin, Heidelberg, 2010. Springer Berlin Heidelberg.

Y. Laguel, W. van Ackooij, J. Malick, and G. M. Ramalho. On the convexity of level-sets of probability functions. 2021. URL `https://api.semanticscholar.org/CorpusID:231846864`.

M. Mokshay. Lecture notes for probabilistic convex geometry: A theory of convexity for probabilistic measures, 2011.

E. Piasini, A. Filipowicz, and J. Levine. Embo: a python package for empirical data analysis using the information bottleneck. URL `https://gitlab.com/epiasini/embo`.

S. Steinert-Thelkeld and J. Szymanik. Ease of learning explains semantic universals. *Cognition*, 195, September 2019.

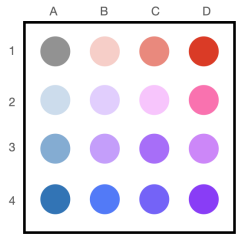
N. Tishby, F. C. Pereira, and W. Bialek. The information bottleneck method. *Proceedings of the 37th Annual Allerton Conference on Communication, Control and Computing*, pages 368–377, September 1999.

N. Zaslavsky, C. Kemp, T. Regier, and N. Tishby. Efficient compression in color naming and its evolution. *PNAS*, 115(31):7937–7942, July 2018.

Appendix A

EXAMPLE APPLICATION OF THE QUASI-CONVEXITY CALCULATION

Suppose we have the following simplified two-dimensional color space and affiliated encoder and we want to determine the encoder's degree of quasi-convexity.



Example Encoder			
	Color Term		
Chip ID	"Red"	"Blue"	"Gray"
A1	0	0	0.9
B1	0	0	0.03
C1	0.2	0	0
D1	0.5	0	0
A2	0	0.1	0.03
B2	0	0.4	0.02
C2	0	0	0.01
D2	0.2	0	0
A3	0	0.1	0
B3	0	0.05	0
C3	0.05	0	0
D3	0.05	0.05	0
A4	0	0.1	0
B4	0	0	0.01
C4	0	0.1	0
D4	0	0.1	0

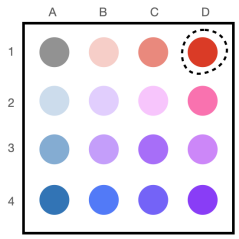
Figure A.1: Left: The example two-dimensional color space comprised of 16 color chips. In this case a chip, c, is considered "in between" two other chips, a and b, if you can draw a straight line between the center of chip a and the center of chip b and it contacts chip c. Right: An example encoder that defines the probability distributions for three different color terms, "red", "blue", and "gray", over the two-dimensional color space.

We begin by calculating the degree of quasi-convexity of the probability distribution affiliated with the "red" color term. We want to determine the convexity of each of the level sets of the probability distribution, starting at the maximum value of $p = 1.0$ and using a mesh of 0.05 to define the jump between level sets. We initialize $qc_{red} = 0$.

We don't have any color chips for which the "red" distribution assigns a probability greater than or equal to p until $p = 0.5$ so for the level sets where $p = 0.95, \ldots, 0.6, 0.55$ we assume $qc_p = 1.0$ and update $qc_{red} = qc_{red} + mesh * qc_{level}$ at each step, meaning when

$p = 0.55$ we have $qc_{red} = 0.45$

In Figure A.2 we see that the first level set, which contains all color chips for which the "red" probability distribution assigned a probability of 0.5 or greater, contains only one chip. The convex hull of a single point is the point itself, so we determine that this level set is perfectly convex, i.e. $qc_{0.5} = 1$. We then update the quasi-convexity value for the "red" color term distribution so that $qc_{red} = qc_{red} + mesh * qc_{0.5} = 0.45 + 0.05 * 1.0 = 0.50$.



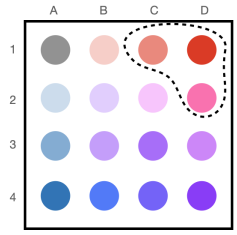
Example Encoder			
	Color Term		
Chip ID	"Red"	"Blue"	"Gray"
A1	0	0	0.9
B1	0	0	0.03
C1	0.2	0	0
D1	0.5	0	0
A2	0	0.1	0.03
B2	0	0.4	0.02
C2	0	0	0.01
D2	0.2	0	0
A3	0	0.1	0
B3	0	0.05	0
C3	0.05	0	0
D3	0.05	0.05	0
A4	0	0.1	0
B4	0	0	0.01
C4	0	0.1	0
D4	0	0.1	0

Figure A.2: The p=0.5 level set of the "red" distribution from the example encoder.

We then move on to the next level set, considering all chips for which the "red" probability distribution assigns a probability of 0.45 or greater. This ends up being the exact same case as before and we get $qc_{0.45} = 1$. The updated quasi-convexity value for the "red" color term distribution is now $qc_{red} = 0.55$.

This same pattern continues for the level sets $p = 0.4, 0.35, 0.3, 0.25$. The updated qc_{red} value after completing those steps is $qc_{red} = 0.75$.

When $p = 0.2$ we get the set of color chips such that the "red" probability distribution assigns probability 0.2 or higher to the chip, shown in Figure A.3. The convex hull of these chips is the set of chips themselves (i.e. we are not missing any chips that are in-between two chips within this set), so we determine that this level set is perfectly convex and thus $qc_{0.2} = 1$. As before, we update the quasi-convexity value for the distribution and set $qc_{red} = 0.8$

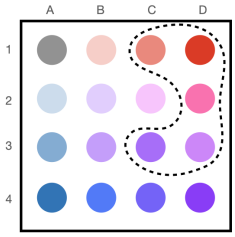


Example Encoder			
	Color Term		
Chip ID	"Red"	"Blue"	"Gray"
A1	0	0	0.9
B1	0	0	0.03
C1	0.2	0	0
D1	0.5	0	0
A2	0	0.1	0.03
B2	0	0.4	0.02
C2	0	0	0.01
D2	0.2	0	0
A3	0	0.1	0
B3	0	0.05	0
C3	0.05	0	0
D3	0.05	0.05	0
A4	0	0.1	0
B4	0	0	0.01
C4	0	0.1	0
D4	0	0.1	0

Figure A.3: The p=0.2 level set of the "red" distribution from the example encoder.

As before, the chips in the level set do not change for $p = 0.15, 0.1$ and the level set is perfectly convex, so we update $qc_{red} = 0.9$.

When $p = 0.05$ the set of color chips contained in the level set expands to include the chips shown in Figure A.4.

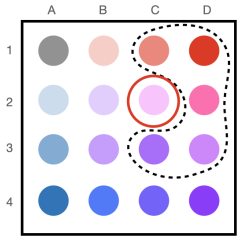


Example Encoder			
	Color Term		
Chip ID	"Red"	"Blue"	"Gray"
A1	0	0	0.9
B1	0	0	0.03
C1	0.2	0	0
D1	0.5	0	0
A2	0	0.1	0.03
B2	0	0.4	0.02
C2	0	0	0.01
D2	0.2	0	0
A3	0	0.1	0
B3	0	0.05	0
C3	0.05	0	0
D3	0.05	0.05	0
A4	0	0.1	0
B4	0	0	0.01
C4	0	0.1	0
D4	0	0.1	0

Figure A.4: The p=0.05 level set of the "red" distribution from the example encoder.

For this particular level set, the convex hull includes an additional color chip that is not included in the level set, as indicated by the red circle in Figure A.5. In this case,

the level set is not perfectly convex, so $qc_{0.05} = \frac{|X_{level}|}{|X_{hull}|} = \frac{5}{6} = 0.833$. We then update the quasi-convexity value for the distribution and set $qc_{red} = 0.9 + 0.05 * 0.833 = 0.9416$.



Example Encoder			
	Color Term		
Chip ID	"Red"	"Blue"	"Gray"
A1	0	0	0.9
B1	0	0	0.03
C1	0.2	0	0
D1	0.5	0	0
A2	0	0.1	0.03
B2	0	0.4	0.02
C2	0	0	0.01
D2	0.2	0	0
A3	0	0.1	0
B3	0	0.05	0
C3	0.05	0	0
D3	0.05	0.05	0
A4	0	0.1	0
B4	0	0	0.01
C4	0	0.1	0
D4	0	0.1	0

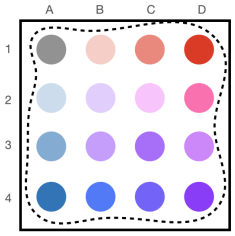
Figure A.5: The p=0.2 level set of the "red" distribution from the example encoder. The missing chip from the convex hull of the level set is circled in red.

With our final iteration, we get $p = 0.0$, meaning the level set includes the entire color space and thus is perfectly convex. We get $qc_{0.0} = 1.0$ and update our quasi-convexity calculation to determine that $qc_{red} = 0.9916$.

We perform similar calculations, shown in Figure A.7 and Figure A.8, to determine $qc_{blue} = 0.973$ and $qc_{gray} = 1.0$.

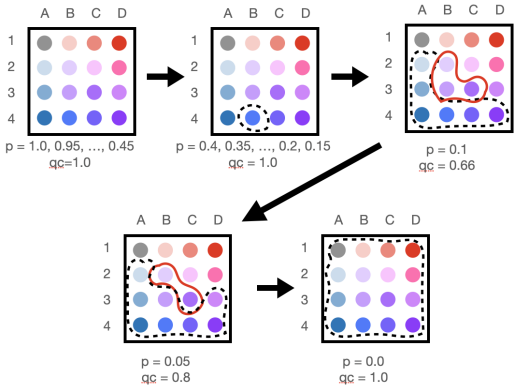
Finally, we determine the degree of quasi-convexity of the overall encoder by taking a weighted sum of the degree of quasi-convexity values of the individual probability distributions that make up the encoder. The weights are determined by looking at how many color chips assign maximal probability to the affiliated color term, as depicted in Figure A.9.

In this case there are 5 out of 16 color chips that assign the "red" color term the maximal weight, 7 out of 16 color chips that assign the "blue" color term the maximal weight, and 4 out of 16 color chips that assign the "gray" color term the maximal weight. Thus, our the encoder's degree of quasi-convexity is calculated as follows:



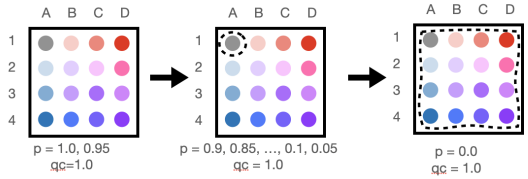
Example Encoder			
	Color Term		
Chip ID	“Red”	“Blue”	“Gray”
A1	0	0	0.9
B1	0	0	0.03
C1	0.2	0	0
D1	0.5	0	0
A2	0	0.1	0.03
B2	0	0.4	0.02
C2	0	0	0.01
D2	0.2	0	0
A3	0	0.1	0
B3	0	0.05	0
C3	0.05	0	0
D3	0.05	0.05	0
A4	0	0.1	0
B4	0	0	0.01
C4	0	0.1	0
D4	0	0.1	0

Figure A.6: The p=0.0 level set of the “red” distribution from the example encoder.



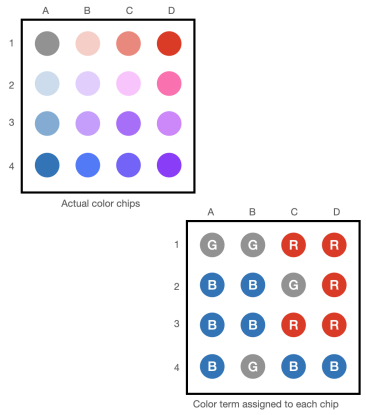
Example Encoder			
	Color Term		
Chip ID	“Red”	“Blue”	“Gray”
A1	0	0	0.9
B1	0	0	0.03
C1	0.2	0	0
D1	0.5	0	0
A2	0	0.1	0.03
B2	0	0.0	0.02
C2	0	0	0.01
D2	0.2	0	0
A3	0	0.1	0
B3	0	0.05	0
C3	0.05	0	0
D3	0.05	0.05	0
A4	0	0.1	0
B4	0	0.4	0.01
C4	0	0.1	0
D4	0	0.1	0

Figure A.7: A depiction of the degree of quasi-convexity calculation applied to the “blue” distribution from the example encoder.



Example Encoder			
	Color Term		
Chip ID	“Red”	“Blue”	“Gray”
A1	0	0	0.9
B1	0	0	0.03
C1	0.2	0	0
D1	0.5	0	0
A2	0	0.1	0.03
B2	0	0.0	0.02
C2	0	0	0.01
D2	0.2	0	0
A3	0	0.1	0
B3	0	0.05	0
C3	0.05	0	0
D3	0.05	0.05	0
A4	0	0.1	0
B4	0	0.4	0.01
C4	0	0.1	0
D4	0	0.1	0

Figure A.8: A depiction of the degree of quasi-convexity calculation applied to the “gray” distribution from the example encoder. Note that because the probabilities defined in the distribution are more granular than the mesh value, the algorithm predicts that this distribution is perfectly convex. A more granular mesh value may yield different results.



Example Encoder			
	Color Term		
Chip ID	“Red”	“Blue”	“Gray”
A1	0	0	0.9
B1	0	0	0.03
C1	0.2	0	0
D1	0.5	0	0
A2	0	0.1	0.03
B2	0	0.4	0.02
C2	0	0	0.01
D2	0.2	0	0
A3	0	0.1	0
B3	0	0.05	0
C3	0.05	0	0
D3	0.05	0.05	0
A4	0	0.1	0
B4	0	0	0.01
C4	0	0.1	0
D4	0	0.1	0

Figure A.9: A depiction of the color term assignments for the color chips, which is used to determine the importance weights for the probability distributions in the encoder’s degree of quasi-convexity calculation.

$$
\begin{aligned}
qc_{encoder} &= wt_{red} * qc_{red} + wt_{blue} * qc_{blue} + wt_{gray} * qc_{gray} \\
&= 0.3125 * 0.9916 + 0.4375 * 0.975 + 0.25 * 1.0 \\
&= 0.9865
\end{aligned}
$$